A Philosophy Basis for Domain Science & Engineering ? Incomplete Draft Sketch

DINES BJØRNER*, Technical University of Denmark, Denmark.

We investigate **a possible philosophy basis** for **domain science & engineering**. There are two bases for this paper: the **philosophy** of **Kai Sørlander** and my work on calculi for the **analysis & description** of **manifest domains,** their **endurants** and **perdurants**. That is, of **parts: natural** and **artifactual** (including components and materials); of **living species: plants** and **animals**, including **humans**; and of the **behaviours** that can be transcendentally deduced from endurants. The philosophy-question to be investigated is "**what must inescapably be in any domain description ?**" that is: "**which are the necessary characteristics of each and every possible world** and **our situation in it.**"

ACM Reference Format:
Dines Bjørner. 2019. **A Philosophy Basis for Domain Science & Engineering ? Incomplete Draft Sketch**. 1, 1 (February 2019), 35 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Draft: February 8, 2019: 09:32 am[1]

1 INTRODUCTION

We investigate a possible philosophy basis for domain science & engineering. The departure point for this investigation is the paper: [14, *A Domain Analysis & Description Method – Principles, Techniques and Modelling Languages*] and Kai Sørlander's books [59, 62, 66].

Characterisation 1. Domain: By a **domain** we shall understand a **rationally describable** segment of a **human assisted** reality, i.e., of the world: its **physical parts** and **living species**. These are **endurants** ("still"), as well as **perdurants** ("alive"). (By 'rational descriptions' we mean: *'in terms of true propositions over primary entities'*[2].) Emphasis is placed on **"human-assistedness"**, that is, that the domain embodies *at least one (man-made)* **artifact** and thus that **humans** are a primary cause for change of endurant **states** as well as perdurant **behaviours**[3] ■[4]

Characterisation 2. Universe of Discourse: The term 'domain', in the context of this paper, is also referred to as 'domain of discourse', or 'universe of discourse'. The term 'universe of discourse' generally refers to the collection of entities (objects) being discussed in a specific discourse. In model-theoretical semantics, a universe of discourse is the set of entities that a model is based on [https://en.wikipedia.org/wiki/Domain_of_discourse] ■

The *science and engineering of domain analysis & description* is *different* from the science of physics and the core of its derived engineerings: building (civil), chemical, mechanical, electrical, electronics, et cetera. All of these engineerings emerged out of *the natural sciences*. These classical engineering disciplines have increasingly included many facets of *man-machine interface* concerns, but their core is still in the *the natural sciences*.

*Fredsvej 11, DK 2840 Holte, Denmark; bjorner@gmail.com, www.imm.dtu.dk/~dibj. Professor emeritus: Techn. Univ. of Denmark DK 2800 Kgs. Lyngby, Denmark

[1] Margin change bars designate major text changes since a February 7, 2019, 17:16 version of this document. This boxed text and margin change bars will be removed when paper is no longer a draft.

[2] The meaning of *'true empirical propositions over primary entities'* will be explained in Sect. 3.1.3 [pp. 17].

[3] We shall, throughout, use the term behaviour in the sense of their being characterisable as sets of sequences of actions, events and behaviours, and **not** from the point of view as propagated by *behaviorism* [58] and `https://en.wikipedia.org/wiki/Behaviorism`, i.e., **not** as a study within the realm of psychology. The two views may not necessarily "disagree".

[4] ■ delimits characterisations and definitions.

Dines Bjørner, DTU Compute, Technical University of Denmark, Richard Petersens Plads, DK 2800 Kgs. Lyngby, Denmark. bjorner@gmail.com.

The core of *domain science & engineering*, such as we shall pursue it, is in two disciplines: *mathematics*, notably *mathematical logic* and *abstract algebra*, and *philosophy*, notably *meta physics* and *epistemology*.

1.1 A Triptych of Software Development

Our focus on *domain analysis & description* stems from the following view of software development: (i) in order to **design software** we must know the *expectations* and *requirements* of users from and of that software; (ii) in order to **prescribe requirements** we must understand its usage *domain* of discourse; and hence, (iii) we must **analyse & describe** that domain – which then implies better **expectation descriptions**. From a formal point of view: $\mathscr{D},\mathscr{S} \models \mathscr{R}$: Based on respective specifications, we can verify (prove, test) that $\mathscr{S}$oftware meets $\mathscr{R}$equirements and expectations in the context of $\mathscr{D}$omain specifications. The essence here is that **we must analyse & describe domain X if we are to have any hope of developing trustworthy software for domain X**.

1.2 The Thesis

The philosophical questions is: *what must inescapably be in any description of a domain ?* With the philosophy of Kai Sørlander, [59, 66], we partially answer that question: *(a)* **space**, *(b)* **time**, *(c)* **physical parts** and *(d)* **living species**. With our investigation of *analysis & description calculi*, [14], we join to the above (partial) answer: *the manifest notions of (e)* **endurant** and *(f)* **perdurant entities**; *(g)* **discrete** and *(h)* **continuous endurants**; *(i)* **artifactual parts** and *(j)* **artifactual materials**; *(k)* **action**, *(l)* **event** and *(m)* **behaviour perdurants**; ; as well as the *internal quality* notions of *(n)* **unique identifiers**, *(o)* **mereologies** and *(p)* **attributes** – such as these, *e–p*, were defined in [14] and are summarised here.

1.3 Some Concepts of Philosophy

Characterisation 3. Philosophy: Philosophy (from Greek "love of wisdom") is the study of general and fundamental problems concerning existence, knowledge, values, reason, mind, and language ■

Characterisation 4. Metaphysics: By *metaphysics* we shall understand a branch of *philosophy* that explores fundamental questions, including the nature of concepts like *being*, *existence*, and *reality*. Traditional metaphysics seeks to answer, in a "suitably abstract and fully general manner", the questions: *What is there ?* and *And what is it like ?* [5] ■

Topics of metaphysical investigation include entities (objects) and their properties, space and time, cause and effect, and possibility.

Characterisation 5. Epistemology: [from epistēmē, 'knowledge', and logos, 'logical discourse'] is the branch of philosophy concerned with the theory of knowledge[6] ■

The philosophy aspect of our study is primarily epistemological, i.e., not metaphysical.

Epistemology studies the nature of knowledge, justification, and the rationality of belief. Much of the debate in epistemology centers on four areas: (1) the philosophical analysis of the nature of knowledge and how it relates to such concepts as truth, belief, and justification, (2) various problems of skepticism, (3) the sources and scope of knowledge and justified belief, and (4) the criteria for knowledge and justification[6].

Observe the distinction in the definitions of metaphysics and epistemology between [metaphysics] *"explores fundamental questions, including the nature of concepts like being, existence, and reality"* and [epistemology] *"the philosophical analysis of the nature of knowledge and how it relates to such concepts as truth, belief, and justification, etc."*. Epistemology addresses such questions as *What makes justified beliefs justified ?*"; *"What does it mean to say that we know something ?"* and, fundamentally, *"How do we know that we know ?"*[6].

[5] https://en.wikipedia.org/wiki/Metaphysics, #Ontology_(Being)
[6] https://en.wikipedia.org/wiki/Epistemology

A central branch of epistemology is *ontology*, the investigation into the basic categories of being and how they relate to one another[7] ■

Characterisation 6. Ontology: By *ontology* we mean the philosophical study of being. More broadly, it studies concepts that directly relate to being, in particular becoming, existence, reality, as well as the basic categories of being and their relations [8] ■

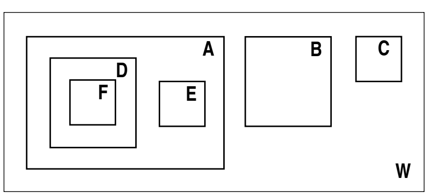
An ontology encompasses a representation, formal naming, and definition of the categories, properties, and relations between the concepts, data, and entities that substantiate one, many, or all domains. Every field creates ontologies to limit complexity and organize information into data and knowledge. As new ontologies are made, their use hopefully improves problem solving within that domain. What ontologies in both information science and philosophy have in common is the attempt to represent entities, ideas, and events, with all their interdependent properties and relations, according to a system of categories. [en.wikipedia.org/wiki/Ontology_(information_science)]

Characterisation 7. Upper Ontology: An upper ontology is a model of the common relations and entities (objects) that are generally applicable across a wide range of domain ontologies. It usually employs a core glossary that contains the terms and associated entity descriptions as they are used in various relevant domain ontologies [en.wikipedia.org/wiki/Ontology_(information_science)] ■

The study of this papers as well as of [14] is a study of upper ontologies. It seems, however, that most ontology studies of the literature focus on endurants (see below). We go well beyond that and also study perdurant components of upper ontology.

Characterisation 8. Mereology: By *mereology* we mean the study of parts, their relations and the wholes they form. Whereas set theory is founded on the membership relation between a set and its elements, mereology emphasizes the meronomic relation between entities, which – from a set-theoretic perspective – is closer to the concept of inclusion between sets. [9] ■

In [16, To Every Manifest Domain a CSP Expression — A Rôle for Mereology in Computer Science] we study mereology in the context of [14]. Our upper ontology focuses on endurants and their mereological relations. Next we present an abstract example:



- The figure shows a mereology.
  - The Whole consists of three parts: A, B and C.
  - Part A is composed from parts D and E.
  - Part D is composite and consists of part F.
  - Parts B, C, E and F are atomic.

We refer to `http://www.columbia.edu/~av72/papers/Space_2007.pdf`: *Spatial Reasoning and Ontology: Parts, Wholes and Locations.* Achille C. Varzi, Columbia University [Published in M. Aiello, I. Pratt-Hartmann, and J. van Benthem (eds.), Handbook of Spatial Logics, Berlin: Springer-Verlag, 2007, pp. 945-1038 (97 pages).]

1.4 Structure of Paper

The paper consists of three main sections. The first two sections, Sect. 2 and Sect. 3, are independent of one another; can thus be read in any order. Sect. 2 brings a summary of the domain analysis & description calculi; Sect. 3 a summary of Kai Sørlander's Philosophy [59, 60, 62, 66]. Section 4 interprets the former in terms of the latter. In Sect. 2 we show examples in footnotes while in Appendix A showing a larger, seven page detailed domain description (of a credit card system).

[7] https://en.wikipedia.org/wiki/Metaphysics#Ontology_(Being)
[8] https://en.wikipedia.org/wiki/Ontology
[9] https://en.wikipedia.org/wiki/Mereology

2 THE DOMAIN ANALYSIS & DESCRIPTION CALCULI

The main sections, Sects. 2.2 and 2.5, of this section are separated by two important sections, Sects. 2.3 and 2.4.

Section 2.1 focus on the broadest analysable & describable notion of **entities**; Sect. 2.2 focus on our notion of **endurants**; Sect. 2.3 then introduces the notion of **transcendental deduction**, a notion which "bridges" endurants with perdurants; Sect. 2.4, as another "aside", covers the inevitable concepts of **space** and **time**; Sect. 2.5 focus on our notion of **perdurants**.

2.1 Entities

Characterisation 9. Entity: By an **entity** we shall understand a **phenomenon**, i.e., something that can be *observe*d, i.e., be seen or touched by humans, *or* that can be *conceive*d as an *abstraction* of an entity; alternatively, a phenomenon is an entity, *if it exists, it is "being", it is that which makes a "thing" what it is: essence, essential nature* [41, Vol. I, pg. 665] ■

We shall, thus, only be concerned with entities. We take the above characterisation of what an entity is to be the same as *a phenomenon* that can be *analysed & described.* And we take that (*analyzable & describable*) as being the same as *analysed & described using the analysis & description prompts outlined in this paper and detailed in [14, A Domain Analysis & Description Method – Principles, Techniques and Modelling Languages].*

We have decided to use the term 'entity'. Some use the term 'object', others the term 'the thing'. Please note that entities are such phenomena which can be analysed & described using the analysis & description prompts introduced in [14] and summarised in this section, i.e., Sect. 2

Characterisation 10. Endurant: By an **endurant** we shall understand an entity that can be observed or conceived and described as a "complete thing" at no matter which given snapshot of time; alternatively an entity is endurant if it is capable of *enduring*, that is *persist*, *"hold out"* [41, Vol. I, pg. 656]. Were we to "freeze" time we would still be able to observe the full endurant ■

Concrete endurants also known as continuants, or in some cases as "substance", are manifest: You can touch them, see them, measure their location and extent as well as many other properties.

Characterisation 11. Perdurant: By a **perdurant** we shall understand an entity for which only a fragment exists if we look at or touch them at any given snapshot in time, that is, were we to freeze time we would only see or touch a fragment of the perdurant, alternatively an entity is perdurant if it endures continuously, over time, persists, lasting [41, Vol. II, pg. 1552] ■

Perdurants also known as occurrents, accidents or happenings, are often what we know as processes, for example: "running". If we freeze time then we only see a part of the running, without any previous knowledge one might not even be able to determine the actual process as being a process of running. Other examples include an activation, a kiss, or a procedure [en.wikipedia.org/wiki/Formal_ontology#Common_terms_in_formal_(upper-level)_ontologies]. The terms 'process'. 'activation' and 'procedure' are here used in their most common sense, i.e., not in their computer science sense.

The terms 'endurant' and 'perdurant' are established in formal ontology [en.wikipedia.org/wiki/Formal_ontology].

2.2 Endurants

We present a terse synopsis of an ontology of endurants, cf. Fig. 1 [next page].

2.2.1 Overview of Endurants:

The observable endurants are either **discrete** or **continuous**, in which latter case we call them **materials**. Discrete endurants are either **physical parts**, or **living species** – **plants** or **animals**, incl. **humans** – or are **structures** (cf. [14, Sect. 3.2.3]).

Physical parts are either **natural parts** or are **artifacts** (man made parts). Artifacts are either sets of **components** – i.e., sets of parts of possibly different sorts[10], or are of **concrete type** – i.e., sets of parts of the same sort [that is, type][11], or are

[10] **Examples:** letters (in a mailbox), goods (in a container), Lego blocks, ...

[11] **Examples:** automobiles (in a fleet of such), links (i.e., street segments of a road net), hubs (street intersections of a road net), ...

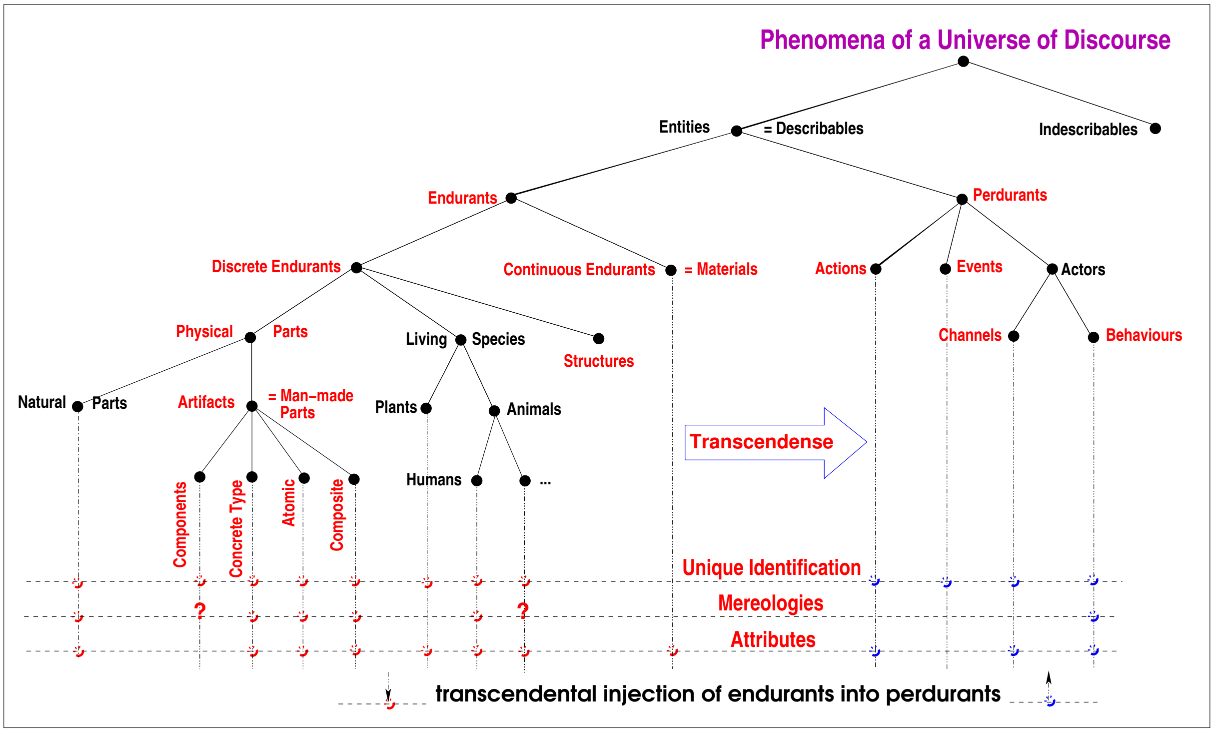


Fig. 1. An Upper Ontology of Domain Entities

atomic artifacts[12] or **composite artifacts**[13]. Materials may contain physical parts and/or living species (`has_parts, ...`). Physical parts may contain materials and/or components (`has_materials, has_components`).

With endurants we shall associate one or more of of the **qualities** (also referred to as **properties**): **unique identifiers**, **mereology**, and **attributes**.

Unique Identifiers: We are mandated to uniquely identify discrete endurants: natural parts[14], man-made parts (artifacts)[15], and living species[16].

Mereology: Endurants that stand in some conceptual or spatial relation to other endurants have that relation captured by the mereology of those endurants. We endow most endurants – natural[17] and man-made[18] parts, and humans[19] – with mereologies.

Space and Time: $\mathbb{S}$pace and $\mathbb{T}$ime are not attributes. They are concepts that, as we shall see, can be deduced transcendentally by rational reasoning. All endurants thus "occur" in both $\mathbb{S}$pace and $\mathbb{T}$ime.

Attributes characterise internal qualities. By attributes we shall mean measurable properties of endurants. Attributes, as we shall later see, are really what we refer to when expressing properties of endurants – or what we shall later refer to as primary objects[20].

---

[12] **Examples:** an automobile, a link, a hub – where we have obviously chosen an abstraction level that "makes" them atomic.
[13] **Examples:** a road net (of hubs, links, ...), a fleet of vehicles (that is, of automobiles, etc.), . . .
[14] **Examples:** the set of diamonds each being uniquely identified.
[15] **Examples:** the set of automobiles together with the set of road links together with a set of road hubs all have their members uniquely identified.
[16] **Examples:** the set of all plants and the set of all animals all have their members uniquely identified.
[17] **Examples:** lakes, rivers and oceans are geographically related, so are the mountain ridges, valleys and flat-lands.
[18] **Examples:** hubs are physically connected to one or more links, links are physically connected to one or two hubs; automobiles are conceptually connectable to all the hubs and links they may pass through; and therefore hubs and links are conceptually connectable to all the automobiles etc.
[19] **Examples:** humans are related genealogically (parents, siblings, etc.), etc.
[20] **Examples:** Attributes of street segments, i.e., links, include length, number of traffic lanes, the history of automobile visits [f.ex., as sequences of $\mathscr{T}$ime-stamped vehicle positions [$\mathscr{S}$patial location] along the link], etc.
Attributes of automobiles include their make, power, length, height, width, weight, and current position: at a hub, or on a link (somewhere), or other.

We reinterpret Michael A. Jackson's [36] *static* and *dynamic: inert*, *reactive* and *active: autonomous*, *biddable* and *programmable* attributes into three categories: *static*, *monitor-able* (*inert*, *reactive*, *autonomous*) and *controllable* (*biddable*, *programmable*).

Artifact Attributes: Focusing on artifacts and their living species human creators – cf. Sect. 3.3.1 [pp. 20] – there arises the possibility, perhaps even the necessity to model attributes of artifacts, attributes that reflect the *intents* of humans when first conceiving these artifacts. Such attributes, in turn, reflect (a) *causality of purpose* (b) *maintenance of their form*, (c) *exchange of matters with an environment*, (d) *purposeful movement*, (e) *sensory organs*, and (f) *instinct*, *incentives*, *feelings*. The above enumeration mirrors essential elements of the contents of Sect. 3.3.1 [pp. 20]. The artifacts that humans build and operate thus possess properties that in turn reflect the above enumeration.

2.2.2 The Analysis and Description Calculi

In [14, *A Domain Analysis & Description Method – Principles, Techniques and Modelling Languages*] we [pedantically] develop two calculi corresponding to the ontology of the previous section. Both are in the form of prompts. These prompts are for the informal use of the domain analyser cum describer.

An Analysis Calculus: "Applying" the prompts of the analysis calculus helps the analyser in deciding whether **phenomena** are *describable* or not, that is, in selecting **entities** for further analysis: `is_entity`. Subsequent prompts, `is_endurant` or `is_perdurant` helps decide whether an entity is one or the other. The full analysis calculus is:

- `is_universe_of_discourse`
- `is_entity`
- `is_endurant`
- `is_perdurant`
- `is_discrete`
- `is_continuous`
- `is_ physical_part`
- `is_living_species`
- `is_structure`
- `is_part`
- `is_atomic`
- `is_composite`
- `is_living_species`
- `is_plant`
- `is_animal`
- `is_human`
- `has_components`
- `has_materials`
- `has_parts`
- `has_living_species`
- `is_artifact`
- `obs_endurant_sorts`
- `has_concrete_type`
- `has_mereology`

A Description Calculus: "Applying" the prompts of the description calculus helps the analyser in deciding, based on the analysis, what descriptions to generate. The full description calculus is:

- `obs_part_sorts`
- `obs_concrete_type`
- `obs_uniq_identifiers`
- `obs_mereology`
- `obs_attributes`
- `obs_component_sorts`
- `obs_material_sorts`

We show some example description schemas below. If an endurant, p, is a composite part then `obs_part_sorts` results in:

Description Schema 1: Composite Parts [Cf. App. A.1.1 [pp. 27]]

Narrative:

- From composite parts p:P
- we can observe parts p_1:P_1, p_2:P_2, ..., p_n:P_n.

Formalisation:

type
- P1, P2, ..., Pn,

value
- obs_Pi: P→Pi, [for i:{1..n}]

If an endurant, p, has a concrete type, T, then `obs_concrete_type` yields:

Description Schema 2: Concrete Parts [Cf. App. A.1.2 [pp. 27]]

Narrative:	Formalisation:
	type
t A concrete type, T, is defined in terms of	t T = $\mathscr{E}$(S1,S2,...,Ss)
s sorts $S_1, S_2, ..., S_s$.	s S1, S2, ..., Ss
o From parts p:P one can observe sets of concrete parts of type T.	**value**
	o obs_T: P → T-**set**

where $\mathscr{E}$(S1,S2,...,Ss) is any type $\mathscr{E}$xpression over sorts S1, S2, ..., Ss.

Appendix A shows a description of the endurants of a simple domain.

Physical parts and living species have unique identifiers. Let p:P be a physical part, $\ell : L$ be a living species.

Description Schema 3: Unique Identifiers [Cf. App. A.2 [pp. 27]]

Narrative:	Formalisation:
	type
• The unique identifier of a part p:P is π:PI and of a living species ℓ:L is ℓi:LI	• PI, LI
	value
* Unique identifiers can be observed.	∗ uid_P: PI, uid_L: LI

Once all unique identifier types and observers have been defined one can state their uniqueness.

Once all unique identifier types and observers have been defined one can define the mereologies of all non-component physical parts and humans. We show the mereology schema for parts.

Description Schema: Mereologies [Cf. App. A.3 [pp. 28]]

Narrative:	Formalisation:
Let p:P be the parts for which we analyse and describe the mereologies.	**type**
* Let $UI_a, UI_b, ..., UI_c$ be the unique identifier types of some arbitrary parts, p_i:P_i, for $i : \{a, b, ..., c\}$, with which parts p:P relate, topologically (i.e., physically) or conceptually.	∗ UI_a, UI_b, ..., UI_c ∗ $\mathscr{M}$(UI_a,UI_b,...,UI_c)
	value
• The mereology of parts p:P is then some set expression, $\mathscr{M}$, over $UI_a, UI_b, ..., UI_c$.	• mereo_P: P → $\mathscr{M}$(UI_a,UI_b,...,UI_c)

Description Schema 5: Attributes [Cf. App. A.4 [pp. 29]]

Narrative:	Formalisation:
	type
t The attributes of parts p:P are $A_1, A_2, ...A_a$.	[t] A1, A2, ... Aa
o They can be observed from parts p:P.	**value**
	[o] attr_Ai: P → Ai, [i:{1..a}]

2.2.3 States [App. A.1.3 [pp. 27]]. By a **state** we shall understand any collection of **physical parts**, **materials**, or **living species** ■

2.2.4 Intentional "Pull". Intentionality is a philosophical concept and is defined by the Stanford Encyclopedia of Philosophy[21] as the power of minds (i.e., humans) to be about, to represent, or to stand for, things, properties and states of affairs.

Characterisation 12. Intentional "Pull": Two or more artifactual parts of different sorts, but with overlapping sets of intents may excert an intentional "pull" on one another. This intentional "pull" may take many forms. Let p_x:X and p_y:Y be two parts of different sorts (X,Y), and with common intent, ι. Manifestations of their common intent must be expressed as properties (i.e., attributes) of the parts and be subject to constraints, and these must be expressed predicatively ■

Endowing a two or more parts with an intention reflects that their human creators have that intention with those parts.[22] In order to fulfill an intention the *intention-sharing* parts must further reflect attributes that serve to "instrument", i.e. "realise", the intention.

Our notion of intentional "pull" is "inspired" by the physical phenomenon of gravitational pull: where the latter is a law of physics, "God-given", and applies to physical parts (including artifacts), the former is a result of human intention(s), and applies to artifacts (only). We shall have more to say about intentional "pull" in Sects. 2.5.6 [pp. 13], 4.4.1 [pp. 23] and 4.4.3 [pp. 23]. Appendix A.4.5 [pp. 30] gives an informal example of the concept of intentional "pull".

2.2.5 Calculi for Analysing & Describing Endurants. In this section, Sect. 2.2, we have studied endurants and "their calculi". In Sect. 2.5 we shall study perdurants. We show how certain kinds of parts can be "morphed" into behaviours. And we show how to derive essential aspects of their structure and signature from the external and internal qualities of endurant parts.

2.3 Transcendental Deduction

It should be clear to the reader that in domain analysis & description we are reflecting on a number of philosophical issues. First and foremost on those of *epistemology*, especially *ontology*. In this section on a sub-field of epistemology, namely that of a number of issues of *transcendental* nature, we refer to [35, Oxford Companion to Philosophy, pp 878–880 1995], [1, The Cambridge Dictionary of Philosophy, pp 807–810, 1995], and [23, The Blackwell Dictionary of Philosophy, pp 54–55 (1998)].

By **transcendental** we shall understand the philosophical notion: **the a priori or intuitive basis of knowledge, independent of experience** ■

A priori knowledge or intuition is central: By *a priori* we mean that it not only precedes, but also determines rational thought.

By a **transcendental deduction** we shall understand the philosophical notion: **a transcendental "conversion" of one kind of knowledge into a seemingly different kind of knowledge** ■

Example 1: Some Transcendental Deductions

We give some intuitive examples of transcendental deductions. They are from the "domain" of programming languages. (a) There is the syntax of a programming language, and there are the programs that supposedly adhere to this syntax (let us refer to the syntax as F). (b) The software tools, *an automatic theorem prover*[23] and *a model checker*, for example `SPIN` [34], that takes a program and some `theorem`, respectively a `Promela` statement, and proves, respectively checks, the program correct with respect the theorem, or the statement. (c) A compiler and an interpreter for any programming language. (d) The software tool, *a syntax checker*, that takes a program and checks whether it satisfies the syntax, including the statically decidable context conditions, i.e., the *statics semantics* – that tool is one of several forms of transcendental deductions; (e) Yes, indeed, any *abstract interpretation* [21, 26] reflects a transcendental deduction.

[21] Jacob, P. (Aug 31, 2010). Intentionality. Stanford Encyclopedia of Philosophy (https://seop.illc.uva.nl/entries/intentionality/) October 15, 2014, retrieved April 3, 2018.

[22] That may be a reason why humans often anthropomorhise, i.e., to to ascribe human form or attributes to an entity.

First these examples show that there are many transcendental deductions. Secondly they show that there is no single-most preferred transcendental deduction.

A transcendental deduction, crudely speaking, is just any "concept" that can be "linked" to another, not by logical necessity, but by logical (and philosophical) possibility !

By **transcendentality** we shall here mean the philosophical notion: the state or condition of being transcendental ■

Example 2: Transcendentality

We can speak of a bus in at least three *senses:*

(i) The bus as it is being `"maintained, serviced, refueled"`;
(ii) the bus as it `"speeds"` down its route; and
(iii) the bus as it `"appears"` (listed) in a bus time table.

The three *senses* are:

(i) as an **endurant** (here a *part*),
(ii) as a **perdurant** (as we shall see a *behaviour*), and
(iii) as an **attribute**[24]

The above example, we claim, reflects *transcendentality* as follows:

(i) We have knowledge of an endurant (i.e., a part) being an endurant.
(ii) We are then to assume that the perdurant referred to in (ii) is an aspect of the endurant mentioned in (i) – where perdurants are to be assumed to represent a different kind of knowledge.
(iii) And, finally, we are to further assume that the attribute mentioned in (iii) is somehow related to both (i) and (ii) – where at least this attribute is to be assumed to represent yet a different kind of knowledge.

In other words: two (i–ii) kinds of different knowledge; that they relate *must indeed* be based on *a priori knowledge*. Someone claims that they relate ! The two statements (i–ii) are claimed to relate transcendentally.[25]

2.4 Space and Time

This section is a necessary prelude to our treatment of perdurants.

Following Kai Sørlander's Philosophy we must accept that space and time are rationally mandated in any domain description. It is, however not always necessary to model space and time. We can talk about space and time; **and** when we do, we must model them.

2.4.1 Space. General:

Mathematicians and physicists model space in, for example, the form of Hausdorf (or topological) space[26]; or a metric space which is a set for which distances between all members of the set are defined; those distances, taken together, are called a metric on the set; a metric on a space induces topological properties like open and closed sets, which lead to the study of more abstract topological spaces; or Euclidean space, due to *Euclid of Alexandria*.

Space Motivated Philosophically: Indefinite Space: We motivate the concept of indefinite space as follows: [66, pp 154] "*The two relations asymmetric and symmetric, by a transcendental deduction, can be given an interpretation:* The relation (spatial) *direction* is asymmetric; and the relation (spatial) *distance* is symmetric. Direction and distance can be

[25]– the attribute statement was "thrown" in "for good measure", i.e., to highlight the issue !
[26]Armstrong, M. A. (1983) [1979]. Basic Topology. Undergraduate Texts in Mathematics. Springer. ISBN 0-387-90839-0.

understood as spatial relations. From these relations are derived the relation *in-between*. Hence we must conclude that *entities exist in space. Space* is therefore an unavoidable characteristic of any possible world" ■

From the direction and distance relations one can derive *Euclidean Geometry*.

Definite Space: By a **definite space** we shall understand a space with a definite metric ■

There is but just one space. It is all around us, from the inner earth to the farthest galaxy. It is not manifest. We can not observe it as we observe a road or a human.

Space Types: The Spatial Value:

1 There is an abstract notion of (definite) $\mathbb{SPACE}$ of further unanalysable points; and
2 there is a notion of $\mathbb{POINT}$ in $\mathbb{SPACE}$.

type
1 $\mathbb{SPACE}$
2 $\mathbb{POINT}$

Space is not an attribute of endurants. Space is just there. So we do not define an observer, `observe_space`. For us, bound to model mostly artifactual worlds on this earth there is but one space. Although $\mathbb{SPACE}$, as a type, could be thought of as defining more than one space we shall consider these isomorphic !

Spatial Observers.

3 A point observer, observe_$\mathbb{POINT}$, is a function which applies to physical parts, p, and yield a point, $\pi : \mathbb{POINT}$.

value
3 observe_$\mathbb{POINT}$: E $\rightarrow$ $\mathbb{POINT}$

Given a notion of $\mathbb{POINT}$s one can then develop a notion of $\mathbb{LOCATION}$s.

2.4.2 Time. General: Concepts of time[27] continue to fascinate thinkers [29, 42, 46–52, 54, 67]. **J.M.E. McTaggart** (1908, [29, 42, 54]) discussed theories of time around the notions of **"A-series":** with concepts like "past", "present" and "future", and **"B-series":** has terms like "precede", "simultaneous" and "follow". **Johan van Benthem** [67] is the standard reference work on the study of time. **Wayne D. Blizard** [22, 1980] relates abstracted entities to spatial points and time. A recent computer programming-oriented treatment is given in [30, **Mandrioli et al.**, 2013].

Time Motivated Philosophically: Indefinite Time: *We motivate the abstract notion of time as follows.* [66, pp 159] *"Two different states[28] must necessarily be ascribed different incompatible predicates. But how can we ensure so ? Only if states stand in an asymmetric relation to one another. This state relation is also transitive. So that is an indispensable property of any world. By a transcendental deduction we say that* primary entities exist in time*. So every possible world must exist in time"* ■

Definite Time: By a **definite time** we shall understand an abstract representation of time such as for example year, month, day, hour, minute, second, et cetera ■

Example 3: Temporal Notions of Endurants

By temporal notions of endurants we mean time properties of endurants, usually modelled as attributes. Examples are: (i) the time stamped link traffic and (ii) the time stamped hub traffic.

[27] **Time:** *(i) a moving image of eternity; (ii) the number of the movement in respect of the before and the after; (iii) the life of the soul in movement as it passes from one stage of act or experience to another; (iv) a present of things past: memory, a present of things present: sight, and a present of things future: expectations.[1, (i) Plato, (ii) Aristotle, (iii) Plotinus, (iv) Augustine].*

[28] States are formally introduced in Sect. 2.2.3 [pp. 7].

Time Values / Time Points. We shall not be concerned with any representation of time points. A "standard" example would be *February 8, 2019: 09:32 am, GMT*. That is, we leave it to the domain analyser cum describer to choose an own representation [30]. Similarly we shall not be concerned with any representation of time intervals.[29]

4 So there is an abstract type $\mathbb{T}ime$,
5 and an abstract type $\mathbb{TI}$: $\mathbb{T}ime\mathbb{I}nterval$.
6 There is no $\mathbb{T}ime$ origin, but there is a "zero" $\mathbb{TI}$me interval.
7 One can add (subtract) a time interval to (from) a time and obtain a time.
8 One can add and subtract two time intervals and obtain a time interval – with subtraction respecting that the subtrahend is smaller than or equal to the minuend.
9 One can subtract a time from another time obtaining a time interval respecting that the subtrahend is smaller than or equal to the minuend.
10 One can multiply a time interval with a real and obtain a time interval.
11 One can compare two times and two time intervals.

type
4 $\mathbb{T}$
5 $\mathbb{TI}$
value
6 $\mathbf{0}$:$\mathbb{TI}$
7 $+,-$: $\mathbb{T} \times \mathbb{TI} \rightarrow \mathbb{T}$
8 $+,-$: $\mathbb{TI} \times \mathbb{TI} \stackrel{\sim}{\rightarrow} \mathbb{TI}$
9 $-$: $\mathbb{T} \times \mathbb{T} \rightarrow \mathbb{TI}$
10 $*$: $\mathbb{TI} \times$ **Real** $\rightarrow \mathbb{TI}$
11 $<,\leq,=,\neq,\geq,>$: $\mathbb{T} \times \mathbb{T} \rightarrow$ **Bool**
11 $<,\leq,=,\neq,\geq,>$: $\mathbb{TI} \times \mathbb{TI} \rightarrow$ **Bool**
axiom
7 $\forall$ t:$\mathbb{T}$ • t+$\mathbf{0}$ = t

Temporal Observers:

12 We define the signature of the meta-physical time observer.

type
12 $\mathbb{T}$
value
12 record_$\mathbb{TIME}$: **Unit** $\rightarrow \mathbb{T}$

The time recorder applies to nothing and yields a time.

Models of Time: Modern models of time, by mathematicians and physicists evolve around spacetime[30] We shall not be concerned with this notion of time. Models of time related to computing differs from those of mathematicians and physicists in focusing on divergence and convergence, zero (Zenon) time and interleaving time [72] are relevant in studies of real-time, typically distributed computing systems. We shall also not be concerned with this notion of time.

Spatial and Temporal Modelling: It is not always that we are compelled to endow our domain descriptions with those of spatial and/or temporal properties. In our experimental domain descriptions, for example, [5, 6, 8, 10–13, 20], we have either found no need to model space and/or time, or we model them explicitly, using slightly different types and observers than presented above.

2.4.3 Whither Attributes ?

Are space and time attributes of endurants ? Of course not ! Space and time surround us. Every endurant is in the one-and-only space we know of. Every endurant is "somewhere" in that space. We represent that 'somewhere' by a point in space. Every endurant point can be recorded. And every endurant point can be time-stamped.

2.5 Perdurants

In reading the early sections of this section, Sect. 2.5, it is important to keep in mind that to parts we shall, generally, and by a transcendental deduction, associate behaviours. So in a sense, behaviours is what we have uppermost in our mind, behaviours is what it is all about.

[29]– but point out, that although a definite time interval may be referred to by number of years, number of days (less than 365), number of hours (less than 24), number of minutes (less than 60)number of seconds (less than 60), et cetera, this is not a time, but a time interval.
[30]The concept of **Spacetime** was first "announced" by Hermann Minkowski, 1907–08 – based on work by Henri Poincaré, 1905–06, https://en.wikisource.org/wiki/Translation:_The_Fundamental_Equations_for_Electromagnetic_Processes_in_Moving_Bodies

2.5.1 Proactive, Active and Passive Parts We shall make a pragmatic distinction between **proactive**, **active** and **passive** parts.

Characterisation 13. Proactive Parts: By a *proactive* part we shall understand a part which when understood transcendentally, as a behaviour, of its own volition, is able to initiate actions usually interacting with other (either proactive or active) behaviours. These actions serve toward fulfillment of part intents ■

Characterisation 14. Active Parts: By an *active* part we shall understand a part which when understood transcendentally, as a behaviour, offers to interact with proactive behaviours. These actions serve toward fulfillment of part intents ■

Characterisation 15. Passive Parts: By a *passive* part we shall understand a part which is not transcendentally deduced into a behaviour ■

It is the domain analyser cum describer who decides whether a part has some (one or more) intents.

Example 4: Proactive, Active and Passive Parts

For a **road transport system** with *links* (street segments), *hubs* (street intersections) and *automobiles*, we may consider automobiles to be *proactive* (as they drive "hither and dither"), and *links* and *hubs* to be *active*, merely observing – as directed by automobiles – that automobiles are passing by ! We can extend the **road transport system** with *automobile clubs* as composite parts from which we can observe *sets of automobiles*. We can then decide to consider these *clubs* as being *passive parts* !

2.5.2 Discrete Behaviours. By a **discrete behaviour** we shall understand a set of sequences of potentially interacting sets of discrete *actions*, *events* and *behaviours* ■

◊ We shall model domain behaviours in terms of $\mathtt{RSL}^+$ *processes*.

We shall not deal with any notion of continuous behaviour.

• • •

We shall also keep in mind that the domains we shall model mostly, as it turns out, focus on man-made parts. These parts have been designed and constructed to serve one or more intents. Usually two or more [separate] parts relate in their intents, for **example:** (a) *roads [are intended to] accommodate automobiles*; (b) *banks and shops [are intended to] accommodate credit cards*; and (c) *canal locks [are intended] to raise and lower watercraft*.

• • •

Perdurants are thus analysed into *actions*, *events*, and *behaviours*. We shall define these terms below. **Common to all of them is that they potentially change a state.** Actions and events are here considered atomic perdurants. For behaviours we distinguish between discrete and continuous behaviours.

2.5.3 Actors: By an **actor** we shall understand something that is capable of initiating and/or **carrying out** actions, events or behaviours ■

Actors will play an important rôle in our *domain analyser & describer*. By what we learn from our study of Sørlander's Philosophy some endurants (of a kind we shall introduce later[31]) can, by a *transcendental deduction*, "become" perdurants some of which thereby "acting" in rôles of *actors*.

◊ We shall model domain actors in terms of $\mathtt{RSL}^+$ *process definitions*.

2.5.4 Discrete Actions: By a **discrete action** [27, 70, 71] we shall understand a **foreseeable** thing which deliberately and potentially changes a well-formed state, in one step, usually into another, still well-formed state, and for which an actor can be made responsible ■

Actions, as we have already noted, serve to fulfill intentions.

[31] *humans* and, although not a concept in [59, 66], their *artifacts*

♢ We shall model domain actions in terms of *RSL*$^+$ *clauses that "update"*[32] *controllable attributes* and in terms of RSL$^+$ (CSP) *output/input*.

We refer to [14, Sect. 9.3 Behaviours, [pp. 50–51]] for examples of behaviour designs.

We shall not deal with any notion of continuous action.

2.5.5 Discrete Events:

There are many notions of events [3, 24, 25, 27, 28, 31, 39, 43, 45, 53]. By a **discrete event** we shall understand some **unforeseen** thing, that is, some 'not-planned-for' "action", one which surreptitiously, non-deterministically changes a well-formed state into another, but usually not a well-formed state, for which no particular domain actor can be made responsible ■

♢ We shall model domain events in terms of RSL$^+$ CSP *input* (from monitorable attributes).

Events, to illustrate, may occur, "to parts", when actions involving these fail to operate properly. For **example:** (a) *automobiles crash on a road*, (b) *a credit card purchase exceeds credit*, and (c) *a canal lock fails to open*.

We shall not deal with any notion of continuous event.

2.5.6 Behaviour Synchronisation and Communication.

We refer to Sect. 2.2.4 [pp. 8]. We assume that the reader is familiar with CSP [32, 33, 55, 56]. Behaviours derived from artifacts serve to fulfill intents. Since two or more artifacts may share an intent there derived behaviours may synchronise and communicate. And since any one artifact may share several intents its derived behaviour may alternate between distinct sub-behaviours. For proactive parts these sub-behaviours are typically "separated" by internal non-determinism, $\sqcap$, whereas for just active parts these sub-behaviours are typically "separated" by external non-determinism, $[]$. We refer to the **sketch behaviour definitions** below.

2.5.7 Translating Parts into Behaviours:

The main *transcendental deduction* of the domain analysis & description method is that of *associating* with each physical part a behaviour.

CSP. The association has these main elements: with part mereologies we associate CSP channels; with part attributes we associate CSP process arguments; and with domain behaviour interaction we associate CSP input/output: ch ?/ch ! v.

Channels: Let *p*:*P* be a physical part with which we have decided to associate a behaviour. Let mereo_P(p) be some set of unique identifiers uids. Then that induces a contribution to a CSP channel declaration:

channel { p_ch[i] | i:UID • i ∈ uids } : MSG_P

where MSG_P is the type of the messages communicated over channel p_ch[...].

Sketch Behaviour Signatures: A schematic part, *p*:*P*, behaviour signature is

value
  p_behav: pi:PI × mereo_P(p) × static_attrs(p) → control_attrs(p)
      → **in** monitor_attrs(p)
      → **in**,**out** { p_ch[i] | i:UID • i ∈ mereo_P(p) } **Unit**

p_behav is our chosen name for the behaviour of *p*. pi is uid_P(p), static_attrs(p) the static attributes, by value, of *p*, control_attrs(p) the control attributes, by value, of *p*, and monitor_attrs(p) typically of the form

- attr_A_{m_1}(p)_ch, attr_A_{m_2}(p)_ch, ..., attr_A_{m_m}(p)_ch,

where *m* is the number of monitor-able attributes, designate these. **Unit** indicates that the behaviour is never-ending.

Sketch Behaviour Definitions:

[32] Our RSL$^+$ descriptions avoid using RSL *variables* and hence assignment. Thus updates are expressed in terms of "passing on" as changed values, controllable attribute parameters in tail-recursive behaviour invocations.

The proactive sketch behaviour definition schema:

```
value
0.  p_behav(pi,m,s)(c) ≡
1.    ...
2.    let intent = select_intents(s,c) in
3.    case intent of
4.      ...
5.      i → let vjs=act_val_beh(s,c,i) in
6.            ‖ { p_ch[j]!v | (v,j) ∈ vjs };
7.            let r=[j↦p_ch[j]?|j•(_,j) ∈ vjs] in
8.            p_behav(pi,m,s)(upd_sta(c,r)) end end
9.      ...
0.    end ... end
```

The proactive sketch behaviour definition: (2.) The arbitrary selection of an intent. (5.) The handling of intent *i*: the determination of information needed to carry out intent *i*; (6.) the parallel communication with active "intent-partners"; (7.) the accumulation of "feedback" results, *r*, from those "partners"; (8.) and the resumption of the p_behaviour.

The active behaviour definition schema is:

```
0.  p_behav(pi,m,s)(c) ≡
1.      ...
2.      let (v,i) = ⌷ { p_ch[i] ? | i:UI • i ∈ m } in
3.      ... ;
4.      p_ch[j] ! msg ;
5.      ... ;
6.      let c′ = upd_state(...)(c_as) in
7.      p_behav(pi,m,s)(c′) end ... end
```

The active definition schema symbolically shows (2.) the possibly non-deterministic external accept of communication *v*alues from some *i*ndexed (other) behaviour; (4.) the possible offering of an output *m*e*s*s*age* to that (i=j) or another behaviour ($i \neq j$); (6.) the likely update of the controllable state; and (7.) the resumption of the p_behaviour.

As behaviours may be both proactive wrt. some intents, and [just] active wrt. other intents, actual behaviour definitions may be combinations of the above.

2.5.8 Translation Schemes:

Here, then, follows a major set of transcendental deductions: of mereologies into CSP [33] channels, of attributes into CSP process, i.e., behaviour arguments, and of parts into CSP processes: their signatures and their definition 'bodies'.

Transcendental Schema 1.

is_composite(e)

```
value
  Translate(p) ≡
      let ui = uid_P(p), me = mereo_P(p),
          sa = stat_attr_vals(p), ca = ctrl_attr_vals(p)
          ST = stat_attr_typs(p), CT = ctrl_attr_typs(p),
          IOR = calc_i_o_chn_refs(p) in
    ≪ value
          𝓜_P: P_UI×MT×ST  CT  IOR  Unit
          𝓜_P(ui,me,sa)(ca) ≡ 𝓑_P(ui,me,sa)(ca)  ≫
          Translate(obs_endurant_sorts_P_1(p))
  ≪ ‖ ≫  Translate(obs_endurant_sorts_P_2(p))
  ≪ ‖ ≫  ...
  ≪ ‖ ≫  Translate(obs_endurant_sorts_P_n(p)) end
```

Transcendental Schema 2.

Concrete is_composite(p): p:Q-set

```
type
  Qs = Q-set
```

value
  qs:Q-**set** = **obs**_Qs(p)
  Translate(qs) ≡ ∥ { Translate(q)|q:Q•q ∈ qs }

Transcendental Schema 3.

is_atomic(p)

value
  Translate(p) ≡
    let ui = **uid**_P(p), me = **mereo**_P(p),
      sa = stat_attr_vals(p), ca = ctrl_attr_vals(p),
      ST = stat_attr_typs(p), CT = ctrl_attr_typs(p),
      IOR = calc_i_o_chn_refs(p) **in**
    ⋘ **value**
      $\mathscr{M}_P$: P_UI×MT×ST PT IOR **Unit**
      $\mathscr{M}_P$(ui,me,sa)(ca) ≡ $\mathscr{B}_P$(ui,me,sa)(ca) ⋙ **end** ■

Transcendental Schema 4.

Core Behaviour

The core processes can be understood as never ending, "tail recursively defined" processes:

$\mathscr{B}_P$: uid:P_UI×me:MT×sa:SA
    → ct:CT
    → **in** in_chns(me) **in**,**out** in_out_chns(me) **Unit**
$\mathscr{B}_P$(ui,me,sa)(ca) ≡
    let (me′,ca′) = $\mathscr{F}_P$(ui,me,sa)(ca) **in** $\mathscr{M}_P$(ui,me′,sa)(ca′) **end**

$\mathscr{F}_P$: P_UI×MT×SA → CT→ **in** in_chns(me) in_out_chns(me) → MT×CT ■

2.6 Closing Section 2

A capsule view of the domain analysis & description calculi has been presented. For details we refer to [14]. We have shown how to derive essential aspects of their structure and signature from the external and internal qualities of endurant parts. But we have not shown how to analyse the domain with respect to the composition of actions, events and behaviours cf. $\mathscr{F}_P$(ui,me,sa)(ca) above !

3 KAI SØRLANDER'S PHILOSOPHY

We shall review an essence of [59, 62, 66]. Kai Sørlander's objective [66, pp 131] *"is to investigate the philosophical question: 'what are the necessary characteristics of each and every possible world and our situation in it'. We can reformulate this question into* **the task of determining the necessary logical conditions for every possible description of the world**."

The next five and a half page requires very concentrated reading.

3.1 The Basis.

In this section we shall mostly quote from [59, 66] – as further supported by [62].

'The world is all that is the case. All that can be described in true propositions' [59] pp, 13, ℓ 2-3. *'In science we investigate how the world is factually'* [59] pp, 13, ℓ 7-8. *'Philosophy puts forward another question. We ask of what could not consistently be otherwise'* [59] pp, 13, ℓ 11-12.' [59] :1 pp 13, ℓ 2–3, 2 pp 13, ℓ 7–8, 3 pp 13, ℓ 11–12 :1,2,3

3.1.1 The Inescapable Meaning Principle.

'It is thus the task of philosophy to determine the **inescapable characteristics** *of the world and our situation in it'* [59] pp, 13, ℓ 13-15. In determining these inescapable characteristic '**we cannot refer to our experience ... since the experience cannot tell us anything that could not consistently be otherwise**.' *'Two demands must be satisfied by the philosophical basis.* **The first is that it must not be based on empirical premises. The other is that it cannot consistently be refuted by anybody under any conceivable circumstances.** *These demands can only be satisfied by one assumption'* [59] pp, 13, ℓ 25-28. We shall refer to this assumption as:

The Inescapable Meaning Principle

- The ***Inescapable Meaning Principle*** [59, pg. 13 ℓ13-pg.14 ℓ1] is the recognition of the mutual dependency between
  – *the meaning of designations and*
  – *the consistency relations between propositions.*

As an example of what "goes into" **the inescapable meaning principle** we bring, albeit from the world of computing, that of the description of the *stack* data type (its values and operations).

The Meaning of Designations

Stacks, narrative

13 **Stacks**, s:S, have elements, e:E;
14 the **empty_S** operation takes no arguments and yields a result stack;
15 the **is_empty_S** operation takes an argument stack and yields a Boolean value result.
16 the **stack** operation takes two arguments: an element and a stack and yields a result stack.
17 the **unstack** operation takes an non-empty argument stack and yields a stack result.
18 the **top** operation takes an non-empty argument stack and yields an element result.

The consistency relations, narrative:

19 an empty_S stack is_empty, and a stack with at least one element is not;
20 unstacking an argument stack, stack(e,s), results in the stack s; and
21 inquiring as to the top of a non-empty argument stack, stack(e,s), yields e.

The meaning of designations, formalisation:

type
13. E, S
value
14. empty_S: **Unit** → S
15. is_empty_S: S → **Bool**
16. stack: E × S → S
17. unstack: S $\stackrel{\sim}{\to}$ S
18. top: S $\stackrel{\sim}{\to}$ E

The consistency relations, formalisation:

is_empty(empty_S()) = **true**
is_empty(stack(e,s)) = **false**
unstack(stack(e,s)) = s
top(stack(e,s)) = e

3.1.2 Necessary and Empirical Propositions.

'**A proposition is necessary** if its truth value depends **only** on the meaning of the designators by means of which it is expressed.' '**A proposition is empirical** if its truth value does **not** so depend.' *'That* **the inescapable meaning principle** *is required – in order to answer the question of how the world*

32

must necessarily be – can be seen from the following.' 'It makes it possible to distinguish between necessary and empirical propositions.' 'An empirical proposition must therefore refer to something ... which exists independently of its designators, and it must predicate something about the thing to which it refers.' The definition *'the world is all that is the case. All that can be described in true propositions.'*,[59] :[1] pp 13, ℓ 16–17; [2] pp 13, ℓ 17–18; [3] pp 13, ℓ 20–21; [4] pp 14, ℓ 26–30; [5] pp 13, ℓ 2–3:[1,2,3,4,5] satisfies **the inescapable meaning principle**. '**That which is described in** `necessary` **propositions is that which is common to [all] possible worlds**. **A concrete world is all that can be described in true** `empirical` **propositions**'[59] pp, 15, ℓ15-18.

3.1.3 Primary Entities [Objects].

'An empirical proposition must refer to an independently existing thing and must predicate something about that thing. On that basis it is then possible to deduce how those entities that can be directly referred to in simple empirical propositions must necessarily be. Those things are referred to as `primary entities`. *A deduction of the* `inevitable characteristics` *of a possible world is thus identical to a deduction of how primary entities must necessarily be'*[59] pp, 15, ℓ23-30.

3.1.4 Two Requirements to the Philosophical Basis.

'Two demands have been put to the philosophical basis for our quest: it must not contain empirical preconditions; and the foundation must not consistently be refuted; and it must not consistently be false' [59] pp, 30, ℓ 6-12. The inescapable meaning principle satisfies this basis [59], pp 30, ℓ 16–28.

3.1.5 The Possibility of Truth.

Where Kant builds on the *contradictory* dichotomy of *das Ding an sich* and *das Ding für uns*, that is, **the possibility of self-awareness**, Kai Sørlander builds on **the possibility of truth** [66, pp 136] *"since the possibility of truth cannot in a consistent manner be denied we can hence assume the* **contradiction principle: 'a proposition and its negation cannot both be true'**." We assume that the contradiction principle is a *necessary truth*. [66, pp 136] "**A necessary truth**, *on one side, follows from the meaning of the designations by means of which it is expressed, and, on the other side and at the same instance, define these designations and their mutual meaning. "*

3.1.6 The Logical Connectives.

Kai Sørlander now defines the logical connectives: *conjunction* ('and', ∧), *disjunction* ('or', ∨), and *implication*. They are all, in a sense, "derived" from *the principle of contradiction: a proposition and the negation of the same proposition cannot both be true.* [62, Chapters I–II] provides a thorough analysis.

3.1.7 Necessity and Possibility.

[66, pp 142] *"A proposition is* **necessarily true**, *if its truth follows from the definition of of the designations by means of which it is expressed; then it must be true under all circumstances. A proposition is* **possibly true**, *if its negation is not necessarily true"*.

3.1.8 Empirical Propositions.

An **empirical proposition** refers to an independently existing entities and predicates something that can be either true or false about the referenced entity. The entities that are referenced in empirical propositions have not been completely characterised by these propositions; they are simply those that can be referenced in empirical propositions.

3.2 Logical Conditions for Describing Physical Worlds

Which are the logical conditions of descriptions of any world ? In [59, 66] Kai Sørlander, through a series of transcendental deductions "unravel" the following logical conditions: (i) symmetry and asymmetry, and (ii) transitivity and intransitivity. From these Kai Sørlander transcendentally deduces: (iii) space: direction, distance, (iv) time: before, after, in-between, etc., (v) states and causality.

3.2.1 Symmetry and Asymmetry.

[66, pp 152] *"[In any world] there can be different* primary entities[33]. *Entity A is different from entity B if A can be ascribed a predicate in-commensurable with a predicate ascribed to B. 'Different from'* is a *symmetric*

[33] [62, Chap. III] investigates this universal fact at length.

predicate. If entity *A* is *identical* to entity *B* then *A* cannot be ascribed a predicate which is in-commensurable with any predicate that can be ascribed to *B*; and then *B* is identical to *A*. *'Equal to'* is a *symmetric predicate*. "

3.2.2 Transitivity and Intransitivity. [66, pp 148] *"If A is identical to B and B is identical to C then A is identical to C with identity then being a transitive relation. The relation different from is not transitive; it is an intransitive relation"*.

3.2.3 Space. [66, pp 154] *"The two relations asymmetric and symmetric, by a transcendental deduction, can be given an interpretation:* The relation (spatial) *direction* is asymmetric; and the relation (spatial) *distance* is symmetric. Direction and distance can thus be understood as spatial relations. From these relations are derived the relation *in-between*. **Hence we must conclude, by a transcendental deduction,** *that* **primary entities exist in space.** *Space is therefore an unavoidable characteristic of any possible world"*. From the direction and distance relations one can derive *Euclidean Geometry*.

3.2.4 States. [66, pp 158-159] *"We must assume that primary entities may be ascribed predicates which are not logically required. That is, they may be ascribed predicates incompatible with predicates which they actually satisfy. For it to be logically possible, that one-and-the-same primary entity can be ascribed incompatible predicates, is only logically possible if any primary entity can exist in different states. A primary entity may be* in one state where it can be ascribed one predicate, and in another state where it can be ascribed another incompatible predicate".

3.2.5 Time. [66, pp 159] *"Two such different states must necessarily be ascribed different incompatible predicates. But how can we ensure so ? Only if states stand in an asymmetric relation to one another. This state relation is also transitive. So that is an indispensable property of any world.* **By a transcendental deduction we say** *that* **primary entities exist in time.** *So every possible world must exist in time"*.

3.2.6 Multiple Entities. On the basis of the above Kai Sørlander can then deduce [62, Chap. II] that any world must exhibit multiple entities.

3.2.7 Causality. [66, pp 162-163] *"States are related by the time relations "before" and "after". These are asymmetric and transitive relations. But how can it be so ? Propositions about primary entities at different times must necessarily be logically independent of one another. This follows from the possibility that a primary entity must necessarily be ascribed different, incompatible predicates at different times. It is therefore logically impossible from the primary entities alone to deduce how a primary entity is at on time point to how it is at another time point. How, therefore, can these predicates supposedly of one and the same entity at different time points be about the same entity ? There can be no logical implication about this !* **Transcendentally**, *therefore, there must be a non-logical implicative between propositions about properties of a primary entity at different times. Such a non-logical implicative must depend on empirical circumstances subject to which the primary entity exists. There are no other circumstances. If the state of a primary entity changes then there must be changes in its "circumstances" whose consequences are that the primary entity changes state. And such 'circumstance-changes' will imply primary entity state changes. We shall use the term 'cause' for a preceding 'circumstance-change' that implies a state change of a primary entity. So now we can conclude that every change of state of a primary entity must have a cause, and that "equivalent circumstances" must have "equivalent effects". This form of implication is called* **causal implication**. *And the principle of implication for* **causal principle**. *So every possible world enjoys the causal principle."* Kant's transcendental deduction is fundamentally built on the the *possibility of self-awareness*. Kai Sørlander's transcendental deduction is fundamentally built on the *possibility of truth*. In Kant's thinking the *causal principle* is a prerequisite for possibility of self-awareness. In this way Kai Sørlander avoids Kant's *solipsism*, i.e., *"that only one's own mind is sure to exist"* a solipsism that, however, flaws Kant's otherwise great thinking.

3.2.8 Kinematics. [66, pp 164–165] *"So primary entities exist in space and time. They must have spatial extent and temporal extent. They must therefore be able to change their spatial properties. Both as concerns form and location. But a spatial change in form presupposes a change in location – as the more fundamental. A primary entity which changes location is said to be in* *movement. A primary entity which does not change location is said to be at rest. The velocity*[34] *of a primary entity expresses the distance and direction it moves in a given time interval. Change in velocity of a primary entity is called its acceleration. Acceleration involves either change in velocity, or change in direction of movement, or both."* So far we have reasoned us to fundamental concepts of *kinematics*.

3.2.9 Dynamics. [66, pp 165-165] *"When we "add" causality" to kinematics we obtain dynamics.We can do so, because primary entities are in time. Kinematics imply that that a primary entity changes when it goes from being at rest to be moving. Likewise when it goes from movement to rest. And similarly, when it accelerates (decelerates). So a primary entity has same state of movement if it has same velocity and moves in the same direction. Primary entities change state of movement if they change velocity or direction. So, combining kinematics and the principle of causality, we can* **[transcendentally] deduce** *that if a primary entity changes state of movement then there must be a* **cause***, and we call that cause a* **force***"*.

3.2.10 Newton's Laws. Newton's First Law: [66, pp 165-166] *"Combining kinematics and the principle of causality, and the therefrom deduced concept of force, we can deduce that any change of movement is proportional*[35] *to the force. This implies that a primary entity which* is not under the influence of an external force will continue in the same state of movement – that is, be at rest or conduction a linear movement at constant velocity. **This is Newton's First Law**". **Newton's Second Law:** [66, pp 166] *"That a certain, non-zero force implies change of movement,* imply that the primary entity must excert a certain *resistance* to that change. It must have what we shall call a certain *mass*.[36] From this it follows that *the change in the state of movement of a primary entity* not only is proportional to the excerted force, but also inversely proportional[37] to the mass of that entity.**This is Newton's Second Law**." **Newton's Third Law:** [66, pp 166-167] *"In a possible world*, the forces that affects primary entities must come from "other" primary entities. Primary entities are located in different volumes of space. Their location may interfere with one another in the sense at least of "obstructing" their mutual movements – leading to clashes. In principle we must assume that even primary entities "far away from one another" obstruct. If they clash it must be with *oppositely directed and equal forces.***This is Newton's Third Law**."

3.2.11 Gravitation and Quantum Mechanics. Mutual Attraction: [66, pp 167-168] *"How can primary entities possibly be the source of forces that influence one another ? How can primary entities at all have a mass*[38] *such that it requires forces to change their state of movement ? The answer must be that primary entities excert a mutual influence on one another – that is there is a mutual attraction."* **Gravitation:** [66, pp 168] *"This must be the case for all primary entities. This must mean that all primary entities can be characterised by a universal mutual attraction: a universal gravitation."* **Finite Propagation – A Gravitational Constant:** [66, pp 168] *"Thus mutual attraction must propagate at a certain, finite, velocity. If that velocity was infinite, then it is everywhere and cannot therefore have its source in concretely existing primary entities. But having a finite velocity implies that there must be a propagational speed limit. It must be a constant of nature."*[39] **Gravitational "Pull":** [66, pp 169-170] *"The nature of gravitational "pull" can be* **[transcendentally] deduced***, basically as follows:* Primary entities must basically consist of elements that attract one another, but which are *stable*, and that is only possible if it is, in principle, *impossible to describe these elementary particles precisely*. If there is a fundamental limit to how these basic particles can be described, then it is also precluded that they can undergo continuous change. Hence there is a basis for stability despite mutual attraction. There must be a foundational limit for how precise these descriptions can be. which implies that the elementary particle as a whole can be described statistically." **Quantum Mechanics:** The rest

[34] Velocity has a *speed* and a *vectorial direction*. *Speed* is a scalar, for example of type `kilometers per hour`. *Vectorial direction* is a scalar structure, for example for a spatial direction consisting of geographical elements: x `degrees North,` y `degrees East` $(x+y=90)$`, and` z `degrees Up or Down` $(0 \leq z \leq 90$`, where, if` $z=90$ `we have that both` x `and` y `are` $0)$.

[35] Observe that we have "only" said: *proportional*, meaning also directly proportional, not whether it is logarithmically, or linearly, or polynomially, or exponentially, etc., so.

[36] *Mass* refers loosely to the amount of *matter* in an entity. This is in contrast to *weight* which refers to the *force* exerted on an entity by *gravity*.

[37] Cf. Footnote 35.

[38] cf. Footnote 36 Pg. 19

[39] Let two entities have respective masses m_1 and m_2. Let the forces with which they attract each other be f_1, respectively f_2. Then the *law of gravitation – as it can be deduced by philosophical arguments* – can be expressed as $f_1 = f_2$. The specific force, expressed using Newton's constant G is $f = G \times m_1 \times m_2 \times r^{-2}$ where r is the distance between the two entities and $G = 6.674 \times 10^{-11} \times m^3 \times kg^{-1} \times s^{-2}$ [m:meter, kg:kilogam s:second] – *as derived by physicists*.

is physics: unification of quantum mechanics and Einstein's special relativity has been done; unification of gravitation with Einstein's general theory of relativity is still to be done. and that is only possible if it is, in principle, *impossible to describe these elementary particles precisely*. If there is a fundamental limit to how these basic particles can be described, then it is also precluded that they can undergo continuous change. Hence there is a basis for stability despite mutual attraction. There must be a foundational limit for how precise these descriptions can be. which implies that the elementary particle as a whole can be described statistically. **Quantum Mechanics:** The rest is physics: unification of quantum mechanics and Einstein's special relativity has been done; unification of gravitation with Einstein's general theory of relativity is still to be done.

A Summary: [66, pp 170-173] "**Philosophy lends to physics its results a necessity that physics cannot give them.** *Experiments have shown that Einstein's results – with propagation limits – indeed hold for this world. Philosophy shows that every possible world is subject to a fixed propagation limit. Philosophy also shows that for a possible world to exist it must be built from elementary particles which cannot be individually described (with Newton's theory).*"

3.3 The Logical Conditions for Describing Living Species

3.3.1 Purpose, Life and Evolution.

Causality of Purpose: [66, pp 174] *"If there is to be the possibility of language and meaning then there must exist primary entities which are not entirely encapsulated within the physical conditions and that these primary entities must be stable and can influence one another. This is only possible if such primary entities are subject to a supplementary causality directed at the future: a* **causality of purpose**." **Living Species:** [66, pp 174-175] *"These primary entities are here called living species. What can be deduced about them ? They must have some form they can be* **developed** *to reach; and which they must be causally determined to* **maintain**. *This development and maintenance must be* **in an exchange of matter with an environment**. . . . *It must be possible that living species occur in one of two forms: one form [1] which is characterised by development, form and exchange, and another form [2] which, additionally, can be characterised by the ability of* **purposeful movement***s.* The first [1] we call *plant*s, the second [2] we call *animal*s." **Animate Entities:** [66, pp 176] *"For an animal to purposefully move around there must be "additional conditions" for such self-movements to be in accordance with the principle of causality: they must have sensory organs sensing among others the immediate purpose of its movement; they must have means of motion so that it can move; and they must have instincts, incentives and feelings as causal conditions that what it senses can drive it to movements"* And all of this in accordance with the laws of physics. **Animal Structure:** [66, pp 177-178] *"Animals, to possess these three kinds of "additional conditions", must be built from special units which have an inner relation to their function as a whole: their purposefulness must be built into their physical building units; that is, as we can now say, their genomes; that is, animals are built from genomes which give them the inner determination to such building blocks for instincts, incentives and feelings. Similar kinds of deduction can be carried out with respect to plants.* **Transcendentally one can deduce basic principles of evolution but not its details**"

3.3.2 Consciousness, Learning and Language.

Consciousness and Learning: [66, pp 180-181] *"The existence of animals is a necessary condition for there being language and meaning in any world.* That there can be *language* means that animals are capable of *developing language*. And this must presuppose that animals can *learn from their experience*. To learn implies that animals can *feel* pleasure and distaste and can *learn*. . . . One can therefore **[transcendentally] deduce** that animals must possess such building blocks whose inner determination is a basis for learning and consciousness." **Language:** [66, pp 181-182] *"Animals with higher social interaction uses signs, eventually developing a language. These languages adhere to the same system of defined concepts which are a prerequisite for any description of any world:* namely the system that philosophy lays bare from a basis of **transcendental deductions** and the *principle of contradiction* and its *implicit meaning theory*".

3.3.3 Humans, Knowledge, Responsibility.

Humans: [66, pp 184] *"A human is an animal which has a language."* **Knowledge:** [66, pp 184] *"Humans must be conscious* of having *knowledge* of its concrete situation, and as such that humans can have knowledge about what they feel, and eventually that humans can know whether what they feel is true or false. Consequently *humans can describe their situation correctly*." **Responsibility:** [66, pp 184] *"In this way one can*

[transcendentally] deduce *that humans* can thus have *memory* and hence can have *responsibility*, be *responsible*. Further deductions lead us into *ethics*."

3.4 Closing Section 3

A capsule view of Kai Sørlander's Philosophy has been presented. The first book in which Kai Sørlander develops his philosophy is [59]. [62] reaffirms that philosophy along a different line. [66] is the most recent of Kai Sørlander's books, treating the same subject, but now along a third and different line. We have mostly, and extensively, quoted from [59, 66].

4 AN INTERPRETATION

We refer to the **thesis** of this paper, Sect. 1.2 [pp. 2]. In this section we shall mark by a ▣ those statements where we conclude that specific description categories are indispensable.

4.1 What versus How to Describe ?

Kai Sørlander's Philosophy does not tell us *what to describe*, but it tells us, so is our interpretation, some fundamentals about *how to analyse & describe*. The *domain analyser cum describer determines what*; the *domain analysis & description method mandates principles, techniques and some tools.*

4.2 Entities

The calculi of Sect. 2 are a consequence of the universal fact that there must be more than one entity, in fact, an indefinite number of entities, in any universe, cf. Sect. 3.2.6 [pp. 18]. Thus the existence of the calculi is motivated on philosophical grounds.

Phenomena which can be analysed & described using the calculi of Sect. 2 are entities. So description of entities are the very core elements of any domain description.

▣ **Entities** *are inescapable elements of any world, i.e., any domain.*

4.2.1 Endurants and Perdurants.

Entities are either endurants or perdurants. This distinction is ours. It is not made in Kai Sørlander's Philosophy.

We are not quite happy with the two characterisations of 'endurants' respectively 'perdurants'. They are taken from [35, Oxford Companion to Philosophy, pp 878–880 1995], [1, The Cambridge Dictionary of Philosophy, pp 807–810, 1995], and [23, The Blackwell Dictionary of Philosophy, pp 54–55 (1998)]. We shall stick to them for want of better. One reason is, "crudely" speaking, that endurants, when "represented" in software "occur" as *data*, while the counterpart of endurant parts (i.e., perdurants) "occur" a 'procedures/processes'. Another reason for making the distinction is that we can then, as a *transcendental deduction*, relate endurants to perdurants

4.3 Endurants

Discussion 1. Discrete and Continuous Endurants: Our distinction between 'discrete' and 'continuous' endurants may be problematic. Why just those two "alternatives" ? As an example: water is deemed 'continuous', so is 'vapour'. One might claim that some vapour is water at one temperature, but that the "same" entity is vapour at another temperature. So should one not have several distinctions instead of one for 'continuous' ? Generally: at one temperature an endurant is judged discrete, while at another, a usually higher, temperature, it is judged continuous. We focus on artifacts and their cause ■

In this paper we do not discuss natural endurants (parts or materials)

Kai Sørlander's Philosophy does not introduce the notions of atomic and composite entities, i.e., parts.

The distinction between atomic and composite parts is important to "our whole story" !

The further analysis into 'components' and 'concrete [set-oriented] parts' is pragmatic and need not, and cannot (?) be justified on philosophical grounds.

4.3.1 Artifactual Endurants.

Kai Sørlander's Philosophy does not specifically make a distinction between natural (as pre-given in the physical world) and artifactual endurants. But his thinking applies to both natural and artifactual endurants.

Discussion 2. The Cause of Artifactual Endurants: *We shall take it as a fact that humans are the cause of artifacts.* Humans conceive, for example, of automobiles and roads, and humans endow these with *intents:* automobiles to drive on roads and roads to accept automobiles.

Hence **the intentional "pull"**: if we can talk of automobiles on roads, then we can describe that as a `traffic` attribute of any automobile; and if we can talk of roads accommodating automobiles, then we can describe that as a `traffic` attribute of any road, and hence we can postulate the intentional pull: The sum total of automobile traffics is commensurate with the sum total of road traffics. If we do talk, then we must describe ■

▣ **Intents** *and* **Intentional "Pull"** *are aspects of descriptions of artifactual entities in the same way that gravitational pull is of natural world descriptions.*

4.3.2 Unique Identifiers

[cf. pp. 5]. We recall that domains consist of indefinite numbers of endurants, cf. Sect. 4.2 [previous page]. Since they are spatial and cannot overlap we can speak of their unique identification.

The next two characterisations are from https://en.wikipedia.org/wiki/Identity_of_indiscernibles#Indiscernibility_of_identicals – see also https://plato.stanford.edu/entries/identity-indiscernible/.

Characterisation 16. Identity of Indiscernibles: The identity of indiscernibles is an ontological principle that states that there cannot be separate entities that have all their properties in common. That is, entities x and y are identical if every predicate possessed by x is also possessed by y and vice versa; to suppose two things indiscernible is to suppose the same thing under two names. It states that no two distinct things (such as snowflakes) can be exactly alike, but this is intended as a metaphysical principle rather than one of natural science. For any x and y, if x and y have all the same properties, then x is identical to y.

- $\forall x \forall y [\forall P(Px \leftrightarrow Py) \rightarrow x = y] \forall x \forall y [\forall P(Px \leftrightarrow Py) \rightarrow x = y]$ ■

Characterisation 17. Indiscernibility of Identicals: If two entities are in fact one and the same, they have all the same properties. For any x and y, if x is identical to y, then x and y have all the same properties.

- $\forall x \forall y [x = y \rightarrow \forall P(Px \leftrightarrow Py)] \forall x \forall y [x = y \rightarrow \forall P(Px \leftrightarrow Py)]$ ■

We shall adopt both characterisations – disregarding any controversies concerning these two principles. We do so since we restrict ourselves to manifest *substances.*[40]

Discussion 3. Representation of Unique Identifiers: We do not prescribe any representation of unique identifiers. It is enough to say that discrete endurants can be uniquely identified – for example by their LOCATION in SPACE (at a given TIME) ■

▣ We therefore argue that **unique identification** *of discrete endurants are an indispensable aspect of any domain description.*

4.3.3 Mereology

[cf. pp. 5]. The English term *'part'* has many possible meanings and is highly ambiguous. The introductory paragraphs of **https://plato.stanford.edu/entries/mereology/** discusses quite a variety (fifteen) of these. Consonant with our upper level ontology parts are *discrete physical endurants.* It is only for such entities (save *structures*) that we define mereologies.

▣ **Mereologies**, *and hence their description, are an indispensable element in any domain description.*

4.3.4 Attributes

[cf. pp. 5]. Endurants are characterised by one or more attributes. That is a universal fact. It is true for any domain.[41] Attributes are what makes it possible to describe endurants Hence, we conclude that *attributes of entities are an inevitable element in any description of the world !*

▣ **Attributes**, *and hence their description, are an indispensable element in any domain description.*

[40] Substance: anything that has mass and takes up space ■

[41] Although we may not always in any specific domain description, ascribe LOCATION attributes to parts, they nevertheless do possess such attributes, and they are all different. Similarly for attributes concerning the **substance** of discrete parts (wood, cement, clay bricks, steel and other metals, plastic, etc.).

4.4 Perdurants

We motivate that perdurants are an inevitable element of any domain and hence also their description, by referring to how Kai Sørlander arrives at kinematics and dynamics, cf. Sects. 3.2.8 [pp. 18]– 3.2.9 [pp. 19]. Kai Sørlander's argument evolves from the transcendental reasoning that leads to the inevitability of *space* and *time* in any world, via the reasoning for *causality* to *kinematics* and *dynamics*. Purely logical reasoning and transcendental implications. Since kinematics and dynamics apply to natural parts they also apply to artifacts, and it is in this way we argue for the inevitability of *actions*, *events* and *behaviours*. There is an added dimension, though. Whereas Kai Sørlander's argument for kinematics and dynamics, evolved purely without reference to living species, we shall have to include reasoning that does include living species in that artifacts can only be understood as having been conceived and created/constructed by humans **and with some intent**.

▣ We therefore argue that **perdurants** over *discrete endurants are an indispensable aspect of any domain description.*

We do observe perdurants in the manifest, artifactual domains that we describe. But we "arrive" at perdurants, not by empirically observing these in the form of actions. events and behaviours. We "arrive", instead, at behaviours by a transcendental deduction, i.e., by "morphing" pro-active and active parts into behaviours, and only then do we describe the composition of part behaviours in terms of observable sets of sequences of actions and events as they interact with other part behaviours. We have not, neither in this paper nor in [14], studied possible analysis calculi for perdurants: specifically how behaviours are composed from actions, events and (other) behaviours.

The reader must keep in mind that we are analysing & describing actions, events and behaviours in manifest, artifactual domains. We are not describing computable actions, events and processes of computers. But we are describing these domain perdurants[42] in the specification language $\texttt{RSL}^+$ – in what may appear as actions, events and processes – and that may explain any possible confusion.

We shall argue for the distinction between actions, events and behaviours.

TO BE WRITTEN

4.4.1 Actions [cf. Sect. 2.5.4 [pp. 12]].
Actions have been studied from a philosophy-point of view, as done in [70, 71, Wilson & Shpall https://plato.stanford.edu/entries/action/ 2012] and in [2, 27].

We refer to Sects. 2.2.4 [pp. 8], 2.5.6 [pp. 13] and 4.4.3.

MORE TO COME

4.4.2 Events [cf. Sect. 2.5.5 [pp. 13]].
Events have likewise been studied from a philosophy-point of view, as done in [3, 24, 25, 27, 28, 31, 39, 43, 45, 53, Casati & Varzi: https://plato.stanford.edu/entries/events/ 2014].

MORE TO COME

4.4.3 Behaviours [cf. Sect. 2.5.2 [pp. 12]].
We have used the term 'behaviour'[43] in the loose sense[44] of their being sets of sequences of actions, events and behaviours and not from the point of view as propagated by *behaviorism* [58]. We have also noted that the two views may not necessarily "disagree". The point being repeated here is that – by behaviours – we are referring to those associated with *proactive* and *active domain parts*, cf. Sect. 2.5.1, Defns. 13– 14 [pp. 12].

We refer to Sects. 2.2.4 [pp. 8], 2.5.6 [pp. 13] and 4.4.1.

5.2.4 [next page] [68, 69]

MORE TO COME

4.5 Further Issues.

TO BE WRITTEN

[42]We refer to the four ♢ marked statements in Sect. 2.5 on Pages 12–13
[43]– as also noted in Footnote 3 [pp. 1]
[44]This loose sense is then made precise in how we model behaviours in terms of CSP processes as expressed in $\texttt{RSL}^+$.

5 CONCLUSION

Physicists analyse & describe *the natural world* as subject to the laws of Newton etc. Primary tools of description are the differential and integral calculi, statistics, probability theory and the like. In *life sciences* we analyse & describe *living species: plants* (botanists) and *animals* (zoologists, medical scientists, etc.). A primary tool of study is that of biology. In *domain science & engineering* we primarily analyse & describe *the artifactual world* and, to some extent, *humans* as they monitor & control artifacts. In [14] and in this paper we have outlined a primary tool of domain study, the domain analysis & description principles, techniques and languages.

5.1 Have We Achieved A Clarification ?

The claim is here being made that before [7, 9, 14, 17] there was no analysis & description method for studying artifactual domains. In that sense we claim to have achieved a clarification of the what the second half of the title of this paper refers to. As to the first half and the tailing question mark, **does domain science & engineering have a basis in philosophy ?** we now claim that there is a basis, and that we have at least made some contribution towards such a basis. It is all very much a first such study. We expect refutations and further clarifications so that the suggested field of study may progress.

5.2 Open Problems / Future Work

We invite the reader to reflect upon the issues of this paper and on those listed here.

5.2.1 Intents and Intentional "Pull". In this paper as well as in [14], we have only very briefly touched upon the issue of '*intents*' as "laid down" in artifactual parts by human designers, as well as the derived notion of *intentional "pull"*. It should be clear that [14] and this paper has only "scratched a surface", that is, that more dedicated studies of *intents* and *intentional "pull"* are needed.

5.2.2 Artifact Attributes. We refer to Sect. 2.2.1 [pp. 6].

MORE TO COME

5.2.3 Transcendental Deduction. It is the first time we have applied a principle of transcendental deduction. We expect to receive critique (positive and, better, negative) on our use of this principle. We also expect to be able better to delineate the scope for use of this principle.

5.2.4 Process Philosophy. If you discerned some hesitancy in my ascribing, in the examples of Sect. 2.3, perdurants to [some] endurants, there is, perhaps, a reason. The issue of transcendental deduction of behaviours from parts is really not that simple. The reader may rightfully say that my claim is a bit too "bold". Some philosophers have thought more deeply on this than I do here. Alfred North Whitehead, in [68, 69], for example, proposes a different way of considering endurancy and perdurancy of entities. We refer to [40, Ivor Leclerc].

5.2.5 Ethics. In [59, 62, 66] and especially in [60] Kai Sørlander also explores issues of ethics as they can be transcendentally deduced. We shall, ourselves, reflect on Kai Sørlander's study in this area as it may relate to issues of ethics in computer science.

MORE TO COME

5.3 Is Philosophy Useless?

It is often said that *"philosophy is useless"*. In this paper we have *"made use of philosophy"*. Hopefully I have not defiled Kai Sørlander's Philosophy.

With philosophy we are dealing with a human endeavour in which it does not make sense to speak of refutable theories. One philosophers claims, although they can be disputed, can not be refuted in the science sense of that concept.

MORE TO COME

6 ACKNOWLEDGMENTS

A first presentation[45] of my work on a possible philosophy basis was at a Victor Ivannikov Memorial Workshop in Yerevan, Armenia, May 3, 2018. The talk there was based on the 89 page report [15][46]. I acknowledge, with thanks and gratitude, Arutyun Avetisyan, Yuri Shoukourian and Vladimir Sahakyan for hosting me.

An October 26, 2018 presentation[47] was at NUS, the National University of Singapore, for which I wrote a first draft of the present paper based on the 95 page research note [15]. I acknowledge, with thanks and gratitude, Chin Wei Ngan, Olivier Danvy and Khoo Siau Cheng for hosting me.

My most serious thanks goes, naturally, to Kai Sørlander, the Danish philosopher whose books [59–66] inspired and keep inspiring me.

REFERENCES

[1] Rober Audi. *The Cambridge Dictionary of Philosophy*. Cambridge University Press, The Pitt Building, Trumpington Street, Cambridge CB2 1RP, England, 1995.
[2] J. L. Austin. *How to do Things with Words*. Harvard University Press, Cambridge, MA, 1962.
[3] Alain Badiou. *Being and Event*. Continuum, 2005. (Lêtre et l'événements, Edition du Seuil, 1988).
[4] Yves Bertot and Pierre Castéran. *Interactive Theorem Proving and Program Development. Coq'Art: The Calculus of Inductive Constructions*. EATCS Series: Texts in Theoretical Computer Science. Springer, 2004.
[5] Dines Bjørner. Domain Models of "The Market" — in Preparation for E–Transaction Systems. In *Practical Foundations of Business and System Specifications (Eds.: Haim Kilov and Ken Baclawski)*, The Netherlands, December 2002. Kluwer Academic Press. *URL:* http://www2.imm.dtu.dk/~dibj/themarket.pdf.
[6] Dines Bjørner. A Container Line Industry Domain. Techn. report, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, June 2007. *URL:* http://www2.imm.dtu.dk/~db/container-paper.pdf.
[7] Dines Bjørner. Domain Science & Engineering – From Computer Science to The Sciences of Informatics, Part I of II: The Engineering Part. *Kibernetika i sistemny analiz*, 2(4):100–116, May 2010.
[8] Dines Bjørner. The Tokyo Stock Exchange Trading Rules. R&D Experiment, Techn. Univ. of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, 2010. *URL:* http://www2.imm.dtu.dk/~db/todai/tse-1.pdf, http://www2.imm.dtu.dk/~db/todai/tse-2.pdf.
[9] Dines Bjørner. Domain Science & Engineering – From Computer Science to The Sciences of Informatics Part II of II: The Science Part. *Kibernetika i sistemny analiz*, 2(3):100–120, June 2011.
[10] Dines Bjørner. A Credit Card System: Uppsala Draft. Technical Report: Experimental Research, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, November 2016. *URL:* http://www.imm.dtu.dk/~dibj/2016/credit/accs.pdf.
[11] Dines Bjørner. Weather Information Systems: Towards a Domain Description. Technical Report: Experimental Research, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, November 2016. *URL:* http://www.imm.dtu.dk/~dibj/2016/wis/wis-p.pdf.
[12] Dines Bjørner. A Space of Swarms of Drones. Research Note, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, December 2017. *URL:* http://www.imm.dtu.dk/~dibj/2017/swarms/swarm-paper.pdf.
[13] Dines Bjørner. What are Documents ? Research Note, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, July 2017. *URL:* http://www.imm.dtu.dk/~dibj/2017/docs/docs.pdf.
[14] Dines Bjørner. A Domain Analysis & Description Method – Principles, Techniques and Modelling Languages. *ACM Trans. on Software Engineering and Methodology*, November 2018. *URL:* http://www.imm.dtu.dk/~dibj/2018/tosem/Bjorner-TOSEM.pdf.
[15] Dines Bjørner. A Philosophy of Domain Science & Engineering – An Interpretation of Kai Sørlander's Philosophy. Research Note, 95 pages, Te chnical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, Spring 2018. *URL:* http://www.imm.dtu.dk/~dibj/2018/philosophy/filo.pdf.
[16] Dines Bjørner. To Every Manifest Domain a CSP Expression — A Rôle for Mereology in Computer Science. *Journal of Logical and Algebraic Methods in Programming*, 1(94):91–108, January 2018. *URL:* http://www2.compute.dtu.dk/~dibj/2016/mereo/mereo.pdf.
[17] Dines Bjørner. Manifest Domains: Analysis & Description. *Formal Aspects of Computing*, 29(2):175–225, Online: July 2016. *URL:* https://doi.org/10.1007/s00165-016-0385-z (doi: 10.1007/s00165-016-0385-z).
[18] Nikolaj Bjørner, Anca Browne, Michael Colon, Bernd Finkbeiner, Zohar Manna, Henny Sipma, and Tomas Uribe. Verifying Temporal Properties of Reactive Systems: A STeP Tutorial. *Formal Methods in System Design*, 16:227–270, 2000.
[19] Nikolaj Bjørner, Ken McMillan, and Andrey Rybalchenko. Higher-order Program Verification as Satisfiability Modulo Theories with Algebraic Data-types. In *Higher-Order Program Analysis*, June 2013. http://hopa.cs.rhul.ac.uk/files/proceedings.html.
[20] Dines Bjørner. Urban Planning Processes. Research Note, Technical University of Denmark, Fredsvej 11, DK-2840 Holte, Denmark, July 2017. *URL:* http://www.imm.dtu.dk/~dibj/2017/up/urban-planning.pdf.
[21] Bruno Blanchet, Patrick Cousot, Radhia Cousot, Laurent Mauborgne Jerome Feret, Antoine Miné, David Monniaux, and Xavier Rival. A static analyzer for large safety-critical software. In *Programming Language Design and Implementation*, pages 196–207, 2003 .
[22] Wayne D. Blizard. A Formal Theory of Objects, Space and Time. *The Journal of Symbolic Logic*, 55(1):74–89, March 1990.

[45] http://ivannikov-ws.org/docs/Bjorner.pdf
[46] http://www.imm.dtu.dk/~dibj/2018/philosophy/filo.pdf
[47] http://www.imm.dtu.dk/~dibj/2018/philosophy/philo.pdf

[23] Nicholas Bunnin and E.P. Tsui-James, editors. *The Blackwell Companion to Philosophy*. Blackwell Companions to Philosophy. Blackwell Publishers, 108 Cowley Road, Oxford OX4 1JF, UK, 1996.
[24] Roberto Casati and Achille Varzi. Events. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Stanford University, spring 2010 edition, 2010.
[25] Roberto Casati and Achille C. Varzi, editors. *Events*. Ashgate Publishing Group – Dartmouth Publishing Co. Ltd., Wey Court East, Union Road, Farnham, Surrey, GU9 7PT, United Kingdom, 23 March 1996.
[26] P. Cousot and R. Cousot. Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints. In *4th POPL: Principles of Programming and Languages*, pages 238–252. ACM Press, 1977.
[27] Donald Davidson. *Essays on Actions and Events*. Oxford University Press, 1980.
[28] F. Dretske. Can Events Move? *Mind*, 76(479-492), 1967. Reprinted in [25, 1996], pp. 415-428.
[29] David John Farmer. *Being in time: The nature of time in light of McTaggart's paradox*. University Press of America, Lanham, Maryland, 1990. 223 pages.
[30] Carlo A. Furia, Dino Mandrioli, Angelo Morzenti, and Matteo Rossi. *Modeling Time in Computing*. Monographs in Theoretical Computer Science. Springer, 2012.
[31] P.M.S. Hacker. Events and Objects in Space and Time. *Mind*, 91:1–19, 1982. reprinted in [25], pp. 429-447.
[32] C.A.R. Hoare. Communicating Sequential Processes. *Communications of the ACM*, 21(8), Aug. 1978.
[33] C.A.R. Hoare. *Communicating Sequential Processes*. C.A.R. Hoare Series in Computer Science. Prentice-Hall International, 1985. Published electronically: http://www.usingcsp.com/cspbook.pdf (2004).
[34] Gerard J. Holzmann. *The SPIN Model Checker, Primer and Reference Manual*. Addison-Wesley, Reading, Massachusetts, 2003.
[35] Ted Honderich. *The Oxford Companion to Philosophy*. Oxford University Press, Walton St., Oxford OX2 6DP, England, 1995.
[36] Michael A. Jackson. *Software Requirements & Specifications: a lexicon of practice, principles and prejudices*. ACM Press. Addison-Wesley, Reading, England, 1995.
[37] Matt Kaufmann, Panagiotis Manolios, and J Strother Moore. *Computer-Aided Reasoning: ACL2 Case Studies*. Kluwer Academic Publishers, June 2000.
[38] Matt Kaufmann, Panagiotis Manolios, and J Strother Moore. *Computer-Aided Reasoning: An Approach*. Kluwer Academic Publishers, June 2000.
[39] Jaegwon Kim. *Supervenience and Mind*. Cambridge University Press, 1993.
[40] Ivor Leclerc. *Whitehead's Metaphysics. An Introductory Exposition*. George Allen and Unwin Ltd and Macmillan, New York 1958, New York, Indiana University Press 1975.
[41] W. Little, H.W. Fowler, J. Coulson, and C.T. Onions. *The Shorter Oxford English Dictionary on Historical Principles*. Clarendon Press, Oxford, England, 1973, 1987. Two vols.
[42] J. M. E. McTaggart. The Unreality of Time. *Mind*, 18(68):457–84, October 1908. New Series. See also: [46].
[43] D.H. Mellor. Things and Causes in Spacetime. *British Journal for the Philosophy of Science*, 31:282–288, 1980.
[44] Tobias Nipkow, Lawrence C. Paulson, and Markus Wenzel. *Isabelle/HOL, A Proof Assistant for Higher-Order Logic*, volume 2283 of *Lecture Notes in Computer Science*. Springer-Verlag, 2002.
[45] Chia-Yi Tony Pi. *Mereology in Event Semantics*. Phd, McGill University, Montreal, Canada, August 1999.
[46] Robin Le Poidevin and Murray MacBeath, editors. *The Philosophy of Time*. Oxford University Press, 1993.
[47] Arthur Prior. *Changes in Events and Changes in Things*, chapter in [46]. Oxford University Press, 1993.
[48] Arthur N. Prior. *Logic and the Basis of Ethics*. Clarendon Press, Oxford, UK, 1949.
[49] Arthur N. Prior. *Formal Logic*. Clarendon Press, Oxford, UK, 1955.
[50] Arthur N. Prior. *Time and Modality*. Oxford University Press, Oxford, UK, 1957.
[51] Arthur N. Prior. *Past, Present and Future*. Clarendon Press, Oxford, UK, 1967.
[52] Arthur N. Prior. *Papers on Time and Tense*. Clarendon Press, Oxford, UK, 1968.
[53] A. Quinton. Objects and Events. *Mind*, 88:197–214, 1979.
[54] Gerald Rochelle. *Behind time: The incoherence of time and McTaggart's atemporal replacement*. Avebury series in philosophy. Ashgate, Brookfield, Vt., USA, 1998. vii + 221 pages.
[55] A. W. Roscoe. *Theory and Practice of Concurrency*. C.A.R. Hoare Series in Computer Science. Prentice-Hall, 1997. *URL:* `http://www.comlab.ox.ac.uk/people/bill.roscoe/publications/68b.pdf`.
[56] Steve Schneider. *Concurrent and Real-time Systems — The CSP Approach*. Worldwide Series in Computer Science. John Wiley & Sons, Ltd., Baffins Lane, Chichester, West Sussex PO19 1UD, England, January 2000.
[57] N. Shankar, S. Owre, J. M. Rushby, and D. W. J. Stringer-Calvert. *PVS Prover Guide*. Computer Science Laboratory, SRI International, Menlo Park, CA, September 1999.
[58] B. F. Skinner. *The behavior of organisms*. Appleton-Century-Crofts, New York, 1938.
[59] Kai Sørlander. *Det Uomgængelige – Filosofiske Deduktioner [The Inevitable – Philosophical Deductions, with a foreword by Georg Henrik von Wright]*. Munksgaard · Rosinante, 1994. 168 pages.
[60] Kai Sørlander. *Under Evighedens Synsvinkel [Under the viewpoint of eternity]*. Munksgaard · Rosinante, 1997. 200 pages.
[61] Kai Sørlander. *Om Menneskerettigheder*. Rosinante, 2000. 171 pages.
[62] Kai Sørlander. *Den Endegyldige Sandhed [The Final Truth]*. Rosinante, 2002. 187 pages.
[63] Kai Sørlander. *Forsvaret for Rationaliteten*. Informations Forlag (Publ.), 2008. 232 pages.
[64] Kai Sørlander. *Den Politiske Forpligtelse – Det Filosofsike Fundament for Demokratisk Stillingtagen*. Informations Forlag, 2011. 280 pages.
[65] Kai Sørlander. *Fornuftens Skæbne – Tanker om Menneskets Vilkår*. Informations Forlag, 2014. 238 pages.
[66] Kai Sørlander. *Indføring i Filosofien [Introduction to The Philosophy]*. Informations Forlag, 2016. 233 pages.
[67] Johan van Benthem. *The Logic of Time*, volume 156 of *Synthese Library: Studies in Epistemology, Logic, Methhodology, and Philosophy of Science (Editor: Jaakko Hintika)*. Kluwer Academic Publishers, P.O.Box 17, NL 3300 AA Dordrecht, The Netherlands, second edition, 1983, 1991.
[68] Alfred North Whitehead. *Science and the Modern World*. The Free Press, New York, 1925.

[69] A.N. Whitehead. *Process and Reality*. Macmillan, New York, 1929.
[70] George Wilson and Samuel Shpall. Action. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Stanford University, summer 2012 edition, 2012.
[71] George Wilson and Samuel Shpall. Action. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Stanford University, summer 2012 edition, 2012.
[72] Wang Yi. *A Calculus of Real Time Systems*. PhD thesis, Department of Computer Sciences, Chalmers University of Technology, Göteborg, Sweden, 1991.

A A DOMAIN ANALYSIS & DESCRIPTION EXAMPLE

The domain is that of a simplified credit card system. It is simplified for example in the following aspects. We model credit card holders in terms of their credit cards. We model the goods (i.e., goods), traded by enterprises, and hence either owned by these or by credit card holders, as (a "pool") separate from both enterprises and credit card holders, i.e., not as elements of composite credit cards or composite enterprises. That is: credit cards, banks, enterprises and goods are considered atomic. From unique goods identifiers we postulate that one can obtain the detailed type of the goods (detailed kind, manufacturer, production date, etc.), but so that all instances of any specific type [still] have unique identification.

MORE TO COME

A.1 Parts

A.1.1 Abstract Sorts: Composite Parts, Page 6

1 A simple credit card system, *SCS*, consists of
2 a conglomerate, *CCs*, of credit cards,
3 a conglomerate, *CBs*, of credit card banks,
4 a conglomerate, *CEs*, of credit card accepting enterprises, and
5 a conglomerate, *CGs*, of goods.

```
type
1 SCS, CCs, CBs, CEs, CGs
value
2 obs_CCs: SCS → CCs
3 obs_CBs: SCS → CBs
4 obs_CEs: SCS → CEs
4 obs_CGs: SCS → CGs
```

A.1.2 Concrete Types: Atomic Parts, Page 7

6 From a conglomerate, *CCs*, of credit cards we can observe a set, *Cs*, of credit cards, *C*.
7 From a conglomerate, *CBs*, of credit card banks we can observe a non-empty set, *Bs*, of banks, *B*.
8 From a conglomerate, *CEs*, of credit card accepting enterprises we can observe a non-empty set, *Es*, of enterprises, *E*.
9 From a conglomerate, *CGs*, of goods we can observe a non-empty set, *Gs*, of goods, *G*.

```
type
6 Cs = C-set, C
7 Bs = B-set, B axiom ∀ bs:Bs•bs≠{}
8 Es = E-set, E axiom ∀ es:Es•es≠{}
9 Gs = G-set, G axiom ∀ gs:Gs•gs≠{}
value
6 obs_Cs: CCs → Cs
7 obs_Bs: CBs → Bs
8 obs_Es: CEs → Es
9 obs_Gs: CGs → Gs
```

A.1.3 States [Sect. 2.2.3 [pp. 7]]:

10 Any value of an entire credit card system is a state.
11 So is the sets of credit cards, banks, enterprises and goods.

```
value
10 ccs:CCS
10 cs:Cs = obs_Cs(obs_CCs(ccs))
10 bs:Bs = obs_Bs(obs_CBs(ccs))
10 es:Es = obs_Es(obs_CEs(ccs))
10 gs:Ps = obs_Gs(obs_CGs(ccs))
```

12 The above definitions can be viewed as a clause of the form:

```
12 ∀ ccs:CCS •
12   let (cs,bs,es,gs)=(obs_Cs,obs_Bs,obs_Es,obs_Gs)(ccs)
12   in ... end
```

that can be bracketed around axiom clauses which refers to cs, bs, es, gs.

A.2 Unique Identifiers, Page 7

We shall omit consideration of identification of conglomerates and sets and focus only on consideration of atomic parts.

A.2.1 Sorts

13 Credit cards have unique identification.
14 Banks have unique identification.
15 Enterprises have unique identification.
16 Goods items have unique identification.
17 The sets of credit card, bank, enterprise and goods [unique] identifiers are disjoint.

```
type
13  CI
14  BI
15  EI
16  GI
value
13  uid_C: C → CI
14  uid_B: B → BI
15  uid_E: E → EI
16  uid_G: G → GI
```

A.2.2 Auxiliary Functions:

From a credit card system we can extract the sets of

18 credit card,
19 bank,
20 enterprise, and
21 goods identifiers.

```
value
18  xtr_CIs: CCS → CI-set
18  xtr_CIs(ccs) ≡ { uid_C(c) | c:C • c ∈ cs }
19  xtr_BIs: CCS → BI-set
19  xtr_BIs(ccs) ≡ { uid_B(b) | b:B • b ∈ bs }
20  xtr_EIs: CCS → EI-set
20  xtr_EIs(ccs) ≡ { uid_E(e) | e:E • c ∈ es }
21  xtr_GIs: CCS → GI-set
21  xtr_GIs(ccs) ≡ { uid_G(g) | g:G • c ∈ gs }
```

A.2.3 Auxiliary Values:

From the universal ccs we can extract the unique identifier constants.

22 all credit card unique identifiers,
23 all bank unique identifiers,
24 all enterprise unique identifiers, and
25 all goods unique identifiers.

```
value
22  cis:CI-set = xtr_CIs(ccs)
23  bis:BI-set = xtr_BIs(ccs)
24  eis:EI-set = xtr_EIs(ccs)
25  gis:GI-set = xtr_GIs(ccs)
```

A.2.4 Part Retrieval Functions

26 Given any credit card identifier we can retrieve the credit card with that identifier.
27 Similarly for bank, enterprise and goods (item) identifiers.

```
value
26  retr_C: CI → C-set ~→ C
26  retr_C(ci)(cs) ≡ ιc:C • c ∈ cs∧uid_C(c)=ci; pre: ci ∈ cis
27  retr_B: BI → B-set ~→ B
27  retr_B(bi)(bs) ≡ ιb:B • b ∈ bs∧uid_B(b)=bi; pre: bi ∈ bis
27  retr_E: EI → E-set ~→ E
27  retr_E(ei)(es) ≡ ιe:E • e ∈ es∧uid_E(e)=ei; pre: ei ∈ eis
27  retr_G: GI → G-set ~→ G
27  retr_G(gi)(gs) ≡ ιg:G • g ∈ gs∧uid_G(g)=gi; pre: gi ∈ gis
```

A.2.5 Constraints

28 All credit card, bank, enterprise and goods (item) identifiers are unique.

```
axiom
28  card cis + card bis + card eis + card gis
28  = card(cis ∪ bis ∪ eis ∪ gis)
```

A.3 Mereologies, Page 7

29 Every credit card (of the system) is conceptually connected to a single bank (of the system), to a subset of the enterprises (of the system) and to [potentially] all goods (of the system).
30 Every bank (...) is conceptually connected to a subset of the credit cards (...) and to a subset of the enterprises (...).
31 Every enterprise (...) is conceptually connected to a subset of the credit cards (...), to a subset of all goods, to a subset of the banks (...), and to a specific bank (...).
32 Every goods (item) (...) is conceptually connected to subsets of the enterprises and of the credit cards (...).

```
type
29  CM = BI × EI-set × GI-set
30  BM = CI-set × EI-set
31  EM = CI-set × BI-set × GI-set × BI
32  GM = EI-set × CI-set
value
29  mereo_C: C → CM
30  mereo_B: B → MM
31  mereo_E: E → EM
32  mereo_G: G → GM
```

A.3.1 (...) Constraints [cf. Sects. A.1.3 and A.2.3]

```
axiom
29  ∀ c:C•c∈cs ⇒
29    let (bi,meis,mpis)=mereo_C(c)
29       ⇒ bi∈bis∧meis⊆bis∧mpis⊆pis end ∧
30  ∀ b:B•b∈bs ⇒
30    let (mcis,meis)=mereo_B(b)
30       ⇒ mcis⊆cis∧meis⊆eis end ∧
31  ∀ e:E•e∈es ⇒
31    let (mcis,mbis,mpis,mbi)=mereo_E(e)
31       ⇒ mcis⊆cis∧mbis∪{mbi}⊆bis∧mpis⊆pis end ∧
32  ∀ g:G•g∈gs ⇒
32    let (geis,gcis)=mereo_G(g)
32       ⇒ geis⊆eis∧gcis⊆cis end
```

A.4 Attributes, Page 7

A.4.1 Credit Cards

33 To every card we associate a possibly empty wish list of desired goods, their supplying enterprise, time when "whished for" and cost – where these supplying enterprises have indeed at that time had that goods (item) for sale and at that price.

34 To every card we associate a possibly empty set of goods – represented by their unique identifiers – "mapped" into a triplet: identity of enterprise, time when acquired and

35 acquisition cost – where these supplying enterprises have indeed at that time had that goods (item) for sale and at that price.

```
type
33 WISH = GI →m (EI×𝕋×COST)
34 ACQ = GI →m (EI×𝕋×COST)
35 COST = Nat; axiom ∀ cost:COST • cost>0
value
33 attr_WISH: C → WISH
34 attr_ACQ: C → ACQ
axiom
34 ∀ ccs:CCS let (eis,gis)=(xtr_EIs,xtr_GIs)(ccs) in ⇒
34   ∀ g:G•g∈gis ⇒
35     let (ei,t,cst)=attr_ACQ(p) in ei∈eis∧ ... end
33 ∧ ∀ cc:CC•cc∈cs let wish = attr_WISH(cc) in
33     ∀ gi:GI•gi ∈ dom wish ⇒
33       let (ei,t,cost)=wish(gi) in ei ∈ eis ∧ ... end end end
```

. . . .

A.4.2 Banks

Banks are characterised by

36 a ledger which records, for every customer – whether a credit card (owner) or an enterprise (owner) – its credit balance; and

37 a cash register which reflects the sum total of all customers' credit balances.

```
type
36 LDGR = (CI|EI)−m>BAL
36 BAL = Intg
37 CASH = Intg
value
36 attr_LDGR: B → LDGR
37 attr_CASH: B → CASH
axiom
  ∀ b:B • b ∈ bs ⇒
36 let (ldgr,cash) = (attr_LDGR,attr_CASH)(b) in
36 dom ldgr ⊆ cis ∪ eis ∧
36 dom ldgr∩cis≠{}∧dom ldgr∩eis≠{}
37 ... [cash] ...
  end
```

. . . .

A.4.3 Enterprises

38 An enterprise records a set of goods as a map from goods (item) identifiers to their price (cost), as well as

39 a record of all most recent sales in the form of a map from goods (item) identifiers to customer (by credit card identifier), sales date (i.e., time) and price (cost),

40 a record of all most recent refunds in the form of a map from goods (item) identifiers to customer (by credit card identifier), return date (i.e., time) and refund, and

41 a history of all transactions in the form a list of enterprise transactions, most recent first.

42 An enterprise transaction is ...

43 Some constraints:

- a goods (for sale) includes returns,
- b sold goods (i.e. sales) are not among the goods for sale, and
- c browsings, sales and returns are recorded (whether successful or not) in the transaction history which is a time stamped sequence of browses, sales and refund transactions.
- d Occurrences of credit card identifications in goods, sales and history attributes must be those of credit card system credit card identifiers.
- e Time stamps in goods (item) histories must be in (for example) descending order. The front history elements' time stamp must be that recorded in sales, respectively refunds.

```
type
38 GS = GI →m COST
39 SS = GI →m (𝕋×CI×COST)
40 RS = GI →m (𝕋×CI×COST)
41 TH = ET*
42 ET = ...
value
38 attr_PS: E → GS
39 attr_SS: E → SS
40 attr_RS: E → RS
41 attr_TH: E → TH
axiom
43 ∀ e:E • e ∈ es ⇒
43     let (gs,ss,rs,th) = (xtr_GS,attr_SS,attr_RS,attr_PH)(e) in
43a    dom rs ⊆ dom gs
43b  ∧ dom ss ∩ dom gs = {}
43c  ∧ ...
43d  ∧ ...
43e  ∧ ... end
```

A.4.4 Goods

44 To every goods (item) we can associate a goods history which, in this case is a sequence of time-stamped (alternative) identifiers of enterprises and credit card (holder)s.

45 Goods are of some kind. Although the unique identifiers of parts are just that, we can postulate a function which from the unique identifiers of goods can ascertain their kind.

46 Goods are further characterised by a number of (static) properties.

47

```
type
```

44 GHist = ($\mathbb{T}$ × (EI|CI))*
45 KIND
46 SP
value
44 attr_GHist: G → GHist
45 attr_KI: G → KI
45 xtr_KI: GI → KI

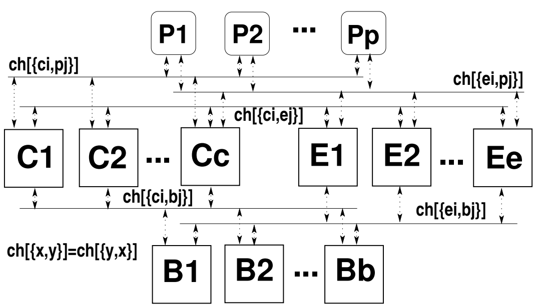
46 attr_SP: G → SP
axiom
46 ∀ g:G • g ∈ gs ⇒
45 attr_KI(g) = xtr_KI(uid_G(g))
46 ∧ [descending time stamps]
46 ∧ [unique identifiers of the system]

A.4.5 Intentional "Pull" [cf. Sect. 2.2.4 [pp. 8]]

48 If a goods (item) is recorded in an enterprise's goods (item) inventory, then somehow it must be recorded as such in that goods history, and vice versa; similarly if recorded in a credit card as acquired.

axiom
48 ([if goods in E's prod. inv., then in goods' hist. as such]
48 ∨ [if goods in C's acq., then in goods' hist. as such])

A.5 Channels [Cf. pp. 13]



49 There are four behaviours.
50 There is one set of channels. Channel indexes are two element sets of unique identifiers of different part types.
51 The mereologies of credit cards and banks are interpreted to mandate channels between credit card behaviours and enterprise behaviours.
52 The mereologies of enterprises and banks are interpreted to mandate channels between enterprise behaviours and bank behaviours.
53 The mereologies of enterprises and goods are interpreted to mandate channels between enterprise behaviours and goods behaviours.
54 The mereologies of credit cards and goods are interpreted to mandate channels between credit card behaviours and goods (item) behaviours.

type
51 CBMsg = ...
?? CEMsg = ...
52 EBMsg = ...
53 EGMsg = ...
54 CPMsg = ...
channel
51 { ch[ci,bj]:CBMsg | ci:CI,bj:BI•ci ∈ cis∧bj ∈ bis }
?? { ch[ci,ej]:CEMsg | ci:CI,ej:EI•ci ∈ cis∧ej ∈ eis }
52 { ch[ei,bj]:EBMsg | ei:EI,bj:BI•ei ∈ eis∧bj ∈ bis }
53 { ch[ei,gj]:EBMsg | ei:EI,gj:GI•ei ∈ eis∧gj ∈ gis }
54 { ch[ci,gj]:EBMsg | ci:CI,gj:GI•ci ∈ cis∧gj ∈ gis }

A.6 Behaviours [Cf. pp. 13]

We shall be using all four transcendental schemes.

55 Using **Transcendental Schema 1** we shall omit (i.e., **skip**) of the **Translate**(p) clause – hence the first 9 lines of the **Translate** clause – which expands into the composition, using **Transcendental Schema 2**, of sets of
56 **Translate**(c_i),
57 **Translate**(b_i),
58 **Translate**(e_i), and
59 **Translate**(p_i).

55 **Translate**(ccs):
55 **Translate**(cs)
55 **Translate**(bs)
55 **Translate**(es)
55 **Translate**(ps)

56 **Translate**(cs): { **Translate**(c) | c:C • c ∈ cs }
57 **Translate**(bs): { **Translate**(b) | b:B • b ∈ bs }
58 **Translate**(es): { **Translate**(e) | e:E • e ∈ es }
59 **Translate**(ps): { **Translate**(p) | p:P • p ∈ ps }

A.6.1 Credit Card Behaviour Signature:

60 The translation of credit card parts yields some `RSL+-text`:
61 the credit_card behaviour signature followed by
62 the credit_card behaviour definition which we, for the moment, omit !

60 **Translate**(c) ≡

61 credit_card: ci:CI×(bi,ceis,cpis):CM×per:PER→WISH×ACQ
61 → **in**,**out** ch[{ci,bi}],
61 {ch[{ci,ei}]|ei:EI•ei∈ceis},
61 {ch[{ci,pi}]|pi:PI•pi∈cpis} **Unit**
62 credit_card(ci,(bi,ceis,cpis),per)(acq) ≡ ...

A.6.2 Bank Behaviour Signature:

63 The translation of bank parts yields some `RSL+-text`:
64 the bank behaviour signature followed by
65 the bank behaviour definition which we, for the moment, omit !

63 **Translate**(b) ≡
64 bank: bi:BI × (bcis,beis):BM × ... → LDGR × CASH
64 → **in**,**out** {ch[{ci,bi}] | ci:CI•ci ∈ bcis},
64 {ch[{bi,ei}] | ei:EI•ei ∈ beis} **Unit**
65 bank(ci,(bcis,beis),...)(ldgr,cash) ≡ ...

A.6.3 Enterprise Behaviour Signature:

66 The translation of enterprise parts yields some RSL$^+$-text:
67 the enterprise behaviour signature followed by
68 the enterprise behaviour definition which we, for the moment, omit !

```
66 Translate(e) ≡
67   enterprise: ei:EI × (ecis,ebis,epis,ebi):EM × ... → PS×SS×RS×TH
67     → in,out {ce_ch[{ci,ei}],eb[{ei,bi}] | ci:CI,bi:BI•ci ∈ ecis∧bi ∈ ebis},
67              {cp_ch[{ei,pi}] | pi:PI • pi ∈ epis} Unit
68   enterprise(ei,(ecis,ebis,epis,ebi),...)(ps,ss,rs,th) ≡ ...
```

A.6.4 Goods Behaviour Signature:

69 The translation of bank parts yields some RSL$^+$-text:
70 the bank behaviour signature followed by
71 the bank behaviour definition which we, for the moment, omit !

```
69 Translate(g) ≡
70   goods: gi:GI × (gcis,geis):PM × pp:PP → GHist
70     → in {cp_ch[{ci,gi}] | ci:CI•ci ∈ gcis}
70          {ep_ch[{ei,gi}] | ei:EI•ei ∈ geis} Unit
71   goods(gi,(gcis,geis),pp)(ghist) ≡ ...
```

A.6.5 System Initialisation:

72 The credit card system behaviour is
73 parallel composition of the set of credit card behaviours in parallel with
74 parallel composition of the set of bank behaviours in parallel with
75 parallel composition of the set of enterprise behaviours in parallel with
76 parallel composition of the set of goods behaviours.

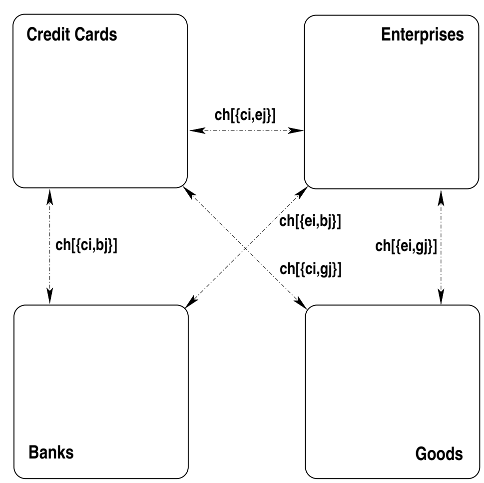
```
72 ccs() ≡
73   ‖ {credit_card(uid_C(c),mereo_C(c),attr_PER(c))(attr_ACQ(c))|c:C•c ∈ cs}
74   ‖ ‖ {bank(uid_B(b),mereo_B(b),...)((attr_LDGR,attr_CASH)(b))|b:B•b ∈ bs}
75   ‖ ‖ {enterprise(uid_E(e),mereo_E(e),...)((attr_PS,attr_SS,attr_RH,attr_PH)(e))|e:E•e ∈ es}
76   ‖ ‖ {goods(uid_G(g),mereo_G(g),...)(attr_GH(g))|g:G•g ∈ gs}
```

A.7 Individual Behaviours

A.7.1 The Intentional Credit Card System Behaviours

The overall intention of the credit card system that we are modelling is that of endowing existing citizens, banks and enterprises with credit cards. Citizens are modelled as credit cards. A citizen purchasing a product (incl. a service) from an enterprise is modelled by the **actions** of the credit card – which is here seen as having a proactive behaviour. We model the following behaviours:

- **Credit card holders** proactively:
  - **browse** (enterprises) for price of products (goods) of specific kinds;
  - **purchase** such goods(from enterprises); and
  - return bought goods (to enterprises) for **refund**s.
- **Banks** actively responding to:
  - inquiries **checking account balances** (from credit cards);
  - **deposit** requests (from enterprises); and
  - **withdrawal** requests (from enterprises).
- **Enterprises** actively responding to (credit card holders):
  - **browsing** queries;
  - **purchase** requests; and
  - **refund** requests.
- **Goods [Products]** actively responding to **move** directives (from enterprises).



Four behaviour clusters and five channel arrays

It may be that some of the channels are not being used in this model. For example the credit card to goods channels: {ch[ci,gj]:EBMsg|ci:CI,gj:GI•ci∈cis∧gj∈gis}.

....

A.7.2 Credit Card Behaviour.

We consider only the proactive facets of credit card[holder]s, not their possible active facets.

77 Credit cards proactively either
78 (window shopping) browses as to desirable products: item, provider and cost,
79 acquires such a product – provided sufficient credit, or
80 returns the product for a refund.

```
value
77 credit_card(ci,(bi,eis,gis),per)(wish,acq) ≡
78     window_shopping(ci,(bi,eis,gis),per)(wish,acq)
79   ⌈⌉ acquisition(ci,(bi,eis,gis),per)(wish,acq)
80   ⌈⌉ refund(ci,(bi,eis,gis),per)(wish,acq)
```

....

In *"window-shopping"* the credit card [holder]

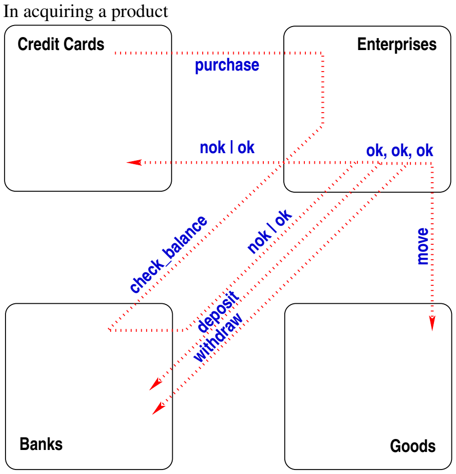
81 selects an arbitrary enterprise, and an arbitrary kind of product; then
82 communicates, ch[ci,ei]!mkWiWh(ci,ki,ti), its desire to browse and, in return, receives a suggestion, i.e., a choice, **let** (ei,picst)=ch[ci,ei]?;
83 if the enterprise has not found an offer
84 then it resumes being an unchanged credit card,
85 else
86 the credit card [holder] updates its *wish list*; and
87 resumes being a credit card with the updated wish list.

type

```
81      KI
value
78     window_shopping(ci,(bi,eis,gis),per)(wish,acq) ≡
81         let ei:EI•ei∈eis, ki:KI in
82,115[48] let re = ch[{ci,ei}]!mkWiSh(ci,ki);ch[{ci,ei}]? in
83         if re="nok"
84            then credit_card(ci,(bi,eis,gis),per)(wish,acq)
85,119       else let (ei,(pi,ti,pri)) = re in
86                 let wish′ = wish † [pi↦(ei,ti,pri)] in
87                 credit_card(ci,(bi,eis,gis),per)(wish′,acq)
78         end end end end end
```

. . . .

In acquiring a product



The [composite] purchase behaviour

88 the credit card [holder] selects a wished-for such, where to buy it and its cost;
89 then it checks with the bank that there are sufficient funds;
90 if there is not sufficient credit the credit card resumes being so with no changes;
91 if there is sufficient credit then a purchase is requested from the enterprise;
92 if the enterprise has the requested item then
  a the following is noted: the purchase time, that the credit card no longer wishes that item, and that the item has now been acquired,
  b whereupon the credit card resumes being so with updated wish and acquisition states,
93 otherwise the resumes being so with no updated states.

```
value
79       acquisition(ci,(bi,eis,gis),...)(wish,acq) ≡
88          let pi:PI•pi∈dom wish in
88          let (ei,ti,pri) = wish(pi) in
89,101   let rb=ch[{ci,bi}]!mkChkBal(pri);ch[{ci,bi}]? end
90,103   if rb=("nok",_)
90           then credit_card(ci,(bi,eis,gis),per)(wish,acq)
91,124       else let re=ch[{ci,ei}]!mkBuy(pi,pri,bi);ch[{ci,ei}]? in
92,127             if re=("ok",ti) then
92a                   let wish′=wish\{pi},
92a                       acq′=acq∪[pi↦(ei,ti,pri)] in
92b                   credit_card(ci,(bi,eis,gis),...)(wish′,acq′) end
93,136            else credit_card(ci,(bi,eis,gis),...)(wish,acq)
79          end end end end end
```

. . . .

In refunding

94 the credit card [holder] selects an acquired product;
95 informs the designated enterprise of the product, its time of acquisition and cost;
96 then the credit card [holder] awaits a response from the enterprise;
97 if that response is "*ok*" then
  a the credit card [holder] removes the refunded product
  b and resumes being a credit card (with an updated acquisition state);
98 else resumes being a credit card (with no changed state).

value

```
80      refund(ci,(bi,eis,gis),per)(wish,acq) ≡
94         let pi:PI•pi∈dom acq in
94         let (ei,t,pri)=acq(pi) in
95,140    ch[{ci,ei}] ! mkRefund(pi,t,pri,bi) ;
96,144    let (r,_)=ch[{ci,ei}] ? in
97         if r="ok" then
97a            let acq′ = acq\{pi} in
97b            credit_card(ci,(bi,eis,gis),per)(wish,acq′) end
98,144    else credit_card(ci,(bi,eis,gis),per)(wish,acq)
80         end end end end
```

. . . .

A.7.3 Bank Behaviour. We here consider only the active facets of a bank, not its possible proactive facets.

[48] The first of the double references in formalisations refer to a narrative description; the second to the narrative+formalisation that "matches", as here, the CSP output clause see Page 33.

A bank

99 external non-deterministically alternates between offering to [synchronise and] communicate with
 a either credit card[holder]s (for checking credit balance) actions;
 b or with enterprises for their deposit actions;
 c or with enterprises for their withdraw actions.

```
99   bank(bi,bm,sa)(ldgr,cash) ≡
99a      check_balance(bi,bm,sa)(ldgr,cash)
99b   ⌈⌉ deposit(bi,bm,sa)(ldgr,cash)
99c   ⌈⌉ withdraw(bi,bm,sa)(ldgr,cash)
```

. . . .

In checking a credit card balance towards some amount (cost) the bank

100 external non-deterministically
101 offers to accept such checking queries from any credit card [holder];
102 retrieves the balance on the designated account;
103 if that balance is less than the designated amount, i.e., cost, then the bank communicates an "*ok*"
104 else a "*nok*" to the inquiring credit card –
105 whereupon it resumes being a bank (with no state change).

```
99a     check_balance(bi,bm,sa)(ldgr,cash) ≡
101,89  ⌈⌉ { let mkChkBal(pri)=ch[{bi,ci}] ? in
102          let bal = ldgr(ci) in
103,90       if pri>bal then ch[{bi,ci}] ! "nok"
104,92                  else ch[{bi,ci}] ! "ok" end
105          bank(bi,bm,sa)(ldgr,cash) end end
100        | ci:CI•ci∈cis }
```

. . . .

In withdrawing an amount of monies from an enterprise holder's account the bank

106 external non-deterministically
107 offers to accept such withdraw communications from its enterprise customers;
108 updates that enterprise holder's (ledger) account
109 whereupon it resumes being a bank (with the ledger state change).

```
99b      withdraw(bi,bm,sa)(ldgr,cash) ≡
107,146  ⌈⌉ { let mkWithdraw(cei,amnt,ti)=ch[{bi,ei}] ?, in
108           let ldgr′=ldgr†[cei↦ldgr(cei)−amnt] in
109           bank(bi,bm,sa)(ldgr′,cash−amnt) end end
106         | ei:EI•ei∈eis }
```

. . . .

In depositing an amount of monies into an enterprise account the bank

110 external non-deterministically
111 offers to accept such deposit communications from its enterprise customers;
112 updates that customers' account with the amount;
113 whereupon it resumes being a bank (with the appropriate state changes).

```
99b      deposit(bi,bm,sa)(ldgr,cash) ≡
111,147  ⌈⌉ { let mkDeposit(cei,amnt,ti)=ch[{bi,ei}] ?, in
112           let ldgr′=ldgr†[cei↦ldgr(cei)+amnt] in
113           bank(bi,bm,sa)(ldgr′,cash+amnt) end end
110         | ei:EI•ei∈eis }
```

. . . .

A.7.4 Enterprise Behaviour. We consider only the active facets of enterprises, not their possible proactive facets.

114 An enterprise externally non-deterministically (⌈⌉)
 a either responds to window browsing,
 b or to purchasing,
 c or to refunds.

```
114   enterprise(ei,em,...)(ps,ss,rh,ph) ≡
114a      browse(ei,em,...)(ps,ss,rh,ph)
114b   ⌈⌉ purchase(ei,em,...)(ps,ss,rh,ph)
114c   ⌈⌉ refund(ei,em,...)(ps,ss,rh,ph)
```

. . . .

In browsing the enterprise

115 offers external non-deterministically, from any credit card [holder], to accept a window shopping inquiry for a product of a designated kind;
116 then, if some (one or more) are for sale,
117 selects an arbitrary product of that kind,
118 retrieves its price, records browsing time,
119 and communicates this to the inquiring credit card [holder];
120 whereupon it resumes being the enterprise with an updated enterprise transaction history.
121 Otherwise
122 the inquiring credit card holder is informed that there is no such product available
123 whereupon the enterprise resumes being the enterprise with an updated enterprise transaction history.

```
114a    browse(ei,em,...)(ps,ss,rh,et) ≡
115,82  ⌈⌉ { let mkWiSh(ci,ki) = ch[{ei,ci}] ? in
117          if ∃ pi:PI•pi ∈ dom ps ∧ xtr_KI(pi)=ki
117            then let pi:PI•pi ∈ dom ps ∧ xtr_KI(pi)=ki in
118              let pri = ps(pi), t = record_𝕋() in
119,85           ch[{ei,ci}] ! mkWiSh(pi,t,pri) ;
120              enterprise(ei,em,...)(ps,ss,rh,⟨(ci,mkWiSh(pi,t,pri))⟩⌢et)
117              end end
121            else
122,83           ch[{ei,ci}] ! "nok" ;
```

```
123          enterprise(ei,em,...)(ps,ss,rh,⟨(ci,mkWiSh(t,"nok"))⟩⌢et)
115       end end
115       | ci:CI • ci ∈ cis }
```

In purchasing the enterprise

124 externally non-deterministically offers to consider a buy order from appropriate credit card holders;
125 the time of purchase is recorded;
126 if the product and buyer-stated price of the order is available at the enterprise and matches its price then five concurrent actions then take place:
127 the buyer is informed of a successful purchase,
128 the buyer's bank is requested to withraw the amount of the price from the cresitvard balance,
129 the enterprise's bank is requested to deposit that amount in the enterprise's account;
130 the product is "relocated", from the enterprise to the buying credit card holder, and
131 the enterprise then resumes being an enterprise now with
132 product removed from being available,
133 updated product sales record,
134 unchanged refund history, and
135 updated transaction history.
136 Otherwise the buying credit card holder is informed of a failed purchase, and
137 the enterprise resumes being an enterprise
138 with no change in products for sale, sales and refund history,
139 but with an updated transaction history.

```
114b    purchase(ei,(ebi,...),...)(ps,ss,rh,th) ≡
124       [] { let mkBuy(gi,pri,cbi) = ch[{ei,cci}] ? ,
125               ti = record_TIME() in
126            if gi ∈ dom ps∧ps(gi)=pri then
127,92          ( ch[{ei,ci}] ! "ok"
128,107         ∥ ch[{ei,cbi}] ! mkWithDraw(ci,pri,ti)
129,111         ∥ ch[{ei,ebi}] ! mkDeposit(ei,pri,ti)
130,151         ∥ ch[{ei,gi}] ! mkOwner(ci,ti)
131             ∥ (enterprise(ei,em,...)
132                    (ps \ {pi},
133                     ss∪[pi↦(ti,ci,pri)],
134                     rh,
135                     ⟨(ti,ci,"ok",mkBuy(gi,pri))⟩⌢th)
126             )       )
136           else ( ch[{ei,ci}] ! "nok" ;
137                  enterprise(ei,em,...)
138                     (ps,
138                      ss,
138                      rh,
139                       ⟨(ti,ci,"nok",mkBuy(gi,pri))⟩⌢th)
125                      )
126            end end
124       | ci:CI • ci ∈ cis }
```

In refunding the enterprise

140 external non-deterministically offers to consider a refund request
141 while recording the refund time;
142 if the returned product was last sold to the customer, at the time and price stated in the refund request,
143 then the refund is [unconditionally] granted resulting in three concurrent actions:
144 notifying the refund requestor of a success refund,
145 notifying the product of its new, "changed-back", ownership, notifying
146 the enterprise's bank to transfer the cost to
147 the credit card holder's bank,
148 while resuming being an enterprise, though with
   a updated products for sale,
   b unchanged "latest" sale,
   c updated latest refund of that product,
   d and updated transaction history.
149 Otherwise the refund is rejected resulting in two concurrent actions:
   a notifying the refund requestor of an unsuccessful refund while
   b resuming being an enterprise,
   c though with only an updated transaction history.

```
114c    refund(ei,(ebi,...),...)(ps,ss,rh,ph) ≡
140,95  [] { let mkRefund(gi,ti,pri,cbi) = ch[{ei,ci}] ?
141                  t = record_TIME() in
142          if gi∈ dom ss ∧ (ti,ci,pri)=ss(gi)
143             then
144,97            ch[{ei,ci}] ! ("ok",t)
146,107         ∥ ch[{ei,ebi}] ! Withdraw(pri,t)
147,111         ∥ ch[{ei,cbi}] ! Deposit(pri,t)
145,151         ∥ ch[{ei,pi}] ! mkOwner(ei,t)
148             ∥ enterprise(ei,em,...)
148a                  (ps∪[gi↦pri],
148b                   ss,
148c                   rh†[gi↦mkRefund(ci,t,pri)],
148d                   ⟨("ok",ci,mkRefund(ci,gi,t,pri))⟩⌢ph)
149             else
144,98            ch[{ei,cci}] ! "nok"
149b            ∥ enterprise(ei,em,...)
149b                  (ps,ss,rh,
149c                   ⟨("nok",ci,mkRefund(gi,t,cst))⟩⌢ph)
142          end end
140       | cci:CCI•cci ∈ cis }
```

A.7.5 Product Behaviour. We consider only the active facets of products, not their possible proactive facets.

The product

150 external non-deterministically offers to accept messages from enterprises
151 informing that product as to its changed ownership;
   a either to henceforth belong to some credit card [holder], where before it belong to an enterprise,
   b or to henceforth belong to some enterprise where before it belong to a credit card [holder]
152 whereupon the product resumes being a product with updated history.

```
150      goods(gi,(eis,cis),...)(gh) ≡
151,145  ⌷ { let mkOwner(i,t) = ch[gi,ei] ? in
151a           assert: i:CI ⇒ ∃ ei:EI•ei∈eis∧hd ph=⟨mkOwner(ei)⟩
151b                 ∧ i:EI ⇒ ∃ ci:CI•ci∈cis∧hd ph=⟨mkOwner(ci)⟩
152            goods(gi,(eis,cis),...)(⟨mkOwner(i,t)⟩⌒ph) end
150          | ei:EI•i∈eis }
```

Contents