Second International Symposium on Engineering Systems
MIT, Cambridge, Massachusetts, June 15-17, 2009

From nutcracking to assisted driving: stratified instrumental systems and the modeling of complexity

Maarten Franssen[1] and Bjørn Jespersen[2]
Delft University of Technology, Delft, the Netherlands

Abstract

Traditional engineering design fails to adequately incorporate into its modeling practice the hybridity *and* stratification *of complexes that involve not only technical artifacts but also individual people playing different roles, as well as social institutions such as laws, norms, and regulations. In this paper we propose a novel way of conceptualizing this complexity. We introduce the notion of* instrumental system*, containing slots for a user, an instrument and an input object, as the central entity of our systems analysis. We identify stratified systems with higher-order systems. A higher-order instrumental system has at least one complex slot, and any or all of its slots may be complex. That a slot is complex means that the sort of entity that fills it is itself systemic. We propose a fairly straightforward and intuitive categorization of the complexity of various instrumental systems. This categorization underpins a taxonomy of instrumental systems, detailing the admissible forms of combination of the three kinds of slot.*

Keywords

Engineering systems, instrumental system, hybridity, complexity, socio-technical system, philosophy of engineering

1. Introduction

Complexity is at the heart of engineering. However, engineers have only recently begun to take note of the concept of complexity, namely, in the form of large-scale complex systems, sometimes called socio-technical systems (see, e.g., [1], [2]). These are systems in which people participate in various capacities at various hierarchical levels, causing the functioning of these systems to depend no less on the actions of these people and the social mechanisms for regulating and coordinating human action than on the smooth mechanical operation of the hardware structures forming the physical backbone of a given system. Engineering is having a

---

[1] Associate Professor, Section of Philosophy, P.O. Box 5015, 2600 GA Delft, m.p.m.franssen@tudelft.nl.

[2] Postdoctoral Researcher, Section of Philosophy, P.O. Box 5015, 2600 GA Delft, b.f.jespersen@tudelft.nl.

hard time modeling these systems. One reason is because engineering has never adequately conceptualized the *mereological complexity* that already crops up in artifacts like nutcrackers. Another reason is because engineering has not succeeded in adequately incorporating into its modeling practice the *hybridity* of complexes involving not only (*i*) technical artifacts but also (*ii*) individual people in their different capacities, together with (*iii*) social institutions such as explicit laws and official standards, and implicit cultural norms.

As a first step in an approach to the modeling of such systems, we propose *instrumental system* as our key notion. Such a system is defined to always contain slots for three different kinds of entity: a *user*, an *instrument*, and an *input object*. The intuitive idea is that a user applies the instrument to the object to transform it from one state to another. The output state is what motivates the application of the tool and also defines the system. Thus, a system boasting a human agent, a nutcracker and a whole nut is a nut-cracking system, and the user’s motive for applying the nutcracker to the whole nut is to obtain a cracked nut. The notion of instrumental system is a *molecular* one. It is not an atomic black box, as this would have made it mysterious how a user, an instrument and an appropriate object would be *related* in such a way as to bring about one activity. The crux of the notion of instrumental system is that it accounts primarily for an *activity*. An instrument, e.g. a nutcracker, is of little relevance on its own and is designed with a user and an input object in mind.

The rest of the paper is organized as follows. Section 2 provides background. Section 3 introduces the notion of instrumental system. Section 4 extends the analysis to low-level nested instrumental systems. Section 5 offers a taxonomy of instrumental systems in terms of which sorts of entities may fill the user, instrument and object spots, respectively.

2. Background

The discipline of systems engineering emerged in the 1940s and 1950s in response to the increasing complexity of the artifacts that engineers were asked to design and develop. The complexity of these artifacts was acknowledged by referring to them as ‘engineering systems’, ‘system’ being the operative word. In the design of these systems, engineers of different backgrounds are involved – mechanical engineers, electro-technical engineers, optical engineers, and so forth – who are each responsible for the design of components – subsystems – of the larger system. The systems engineer is responsible for ensuring the coordinated behavior of these components. Due to the highly complicated character of these systems, this coordination can often be achieved only by delegating the coordination task to human operators. Humans, therefore, are involved in two ways with engineering systems: as users of the system as a whole, and as users of some of its subsystems. The users of the subsystems do not use these for their own benefit, but in order to allow the end-users to use the whole system for their benefit. In the modeling of these systems as practiced by systems engineering, end-users did not figure at all, while operators were conceived as systemic components, on a par with the hardware components and modeled in the same way, as causal mechanisms or algorithmic input/output devices. This view of operators ignored completely the fact that they are intentional human beings endowed with a particular perspective on their role as system operator, a role that is but one role among the many other roles making up their life.

Neglecting the end-user leads to similar difficulties, both because users are also intentional human beings who may well not use the system as intended by its designers, and because many complex engineering systems have many simultaneous users (just think of a road). The actions of fellow users modify the system that any individual user is using, which impacts both upon the objective condition of the system (e.g., whether it is clogged) and its subjective condition relative to each user (i.e., how they individually perceive the system and their own role in it). Since both operators and users are intentional human beings who also act in the light of their individual interests, the analysis of the behavior of a system in its entirety cannot do without the concept of intentionality. The fact that human operators have a mind of their own may be a liability, due to erratic or self-serving behavior that is detrimental to the proper functioning both of the subsystem and overall system, but it may just as likely be an asset. Operators with their wits about them, and perhaps their heart in the right place, can intervene and save the day where the (sub-)system would have otherwise failed.

A growing awareness of the inadequacy of the received view on how to model complex engineering systems that involve people as well as hardware components, as (quasi-)physical systems exclusively, led to the development of so-called soft-systems thinking in the 1960s and 1970s. The original systems-engineering view then came to be known as ‘hard-systems thinking’ (see e.g. [3]). Soft-systems thinking, however, essentially copied the deficiencies of hard-systems thinking by, as it were, changing black into white and white into black. In soft-systems thinking only the intentional relations between the people embedded in a particular system are modeled, and the hardware that these people were operating and trying to make work was considered irrelevant and left out. The upshot, perhaps not surprisingly, was that the approach failed to make much of an impact on the practice of designing, implementing and maintaining large-scale complex engineering systems.

In this paper we propose a novel conceptualization of large-scale, complex engineering systems, a synthesis of the hard-systems and soft-systems approaches outlined above. Any philosophical account of engineering systems must meet the constraints that both the physical and the intentional must be included and their interaction made explicit. We take the *physico-intentional hybridity* that makes up the core of technology – which is to intentionally put physical mechanisms discovered in nature and purpose-built artifacts to human use – as the cornerstone of our account of complex systems.

The hybridity of complex technological systems has been recognized before us, to be sure. Hubka and Eder’s conception of a *transformation system* [4], for example, is close to our notion of an instrumental system. Hubka and Eder, however, acknowledge the hybridity of transformation systems only at the fundamental level and focus their further analysis on just one component of transformation systems, the component they call technical system, the role of which is comparable to the position of instrument in our instrumental systems. Their conception of technical systems is a purely ‘mechanical’ one, in marked contrast to how we conceive of the instrument constituent of instrumental systems. In our view systems in technology are hybrid through and through, and our conception of instrumental systems aims to capture this, as we explain in the next two sections.

A source of inspiration of our notion of instrumental system would be Hughes [5], who also emphasizes the interaction between physical artifacts and human players like organizations. We agree with him that

> "Systems nestle hierarchically like a Russian Easter egg into a pattern of systems and subsystems." (p. 54)

However, we place less emphasis on the embedding environment than does Hughes. For our purposes, the environment is simply the reservoir which an instrumental system recruits its input from and feeds its output back into.

Our approach is not provoked exclusively by a dissatisfaction with the conceptualization that engineering is capable of providing. Also philosophy of technology has so far failed to address the question what engineered systems are. A case in point would be the little that Bunge, one of the few philosophers to address the systemic structure of physical as well as social entities, has to say about what he dubs 'technosystems' ([6], p. 202). He introduces technosystems as a special sort of social system, consisting partly of humans and partly of artifacts. His general characterization of social systems, however, has them consist exclusively of humans; all (other) material objects are part of the environment of social systems. This testifies to the difficulty that also philosophers have in accommodating human beings and the material objects they interact with within a single notion of system.

We proceed in three steps. First, we introduce the notion of instrumental system as the basic entity of our systems analysis. Second, we sketch how to link instrumental systems together to form systems of arbitrary complexity. The simplest instrumental systems turn out to be the simplest complex systems, and much more complex systems can be built up from interlocking instrumental systems. Thirdly and finally, we put forward a taxonomy of admissible manners of filling in the user, instrument and object slots, respectively.

Note that when mentioning instruments like nutcrackers and automobiles in the examples below, we are not offering a theory of nutcrackers and automobiles, but of the sort of instrumental systems that artifacts like these are systemic components of. An analysis of nutcrackers and automobiles would focus on what their respective designed functions were: cracking nutshells, transporting someone or something by means of a motorized vehicle. Here we take their functions for granted and focus instead on their interface and interdependence with intentional users. We thus focus on the two-way fact that, e.g., nutcrackers can crack nutshells only if manipulated correctly by an intentional agent and intentional agents can crack nutshells in a regular (non-improvised, non-*ad hoc*) manner only if using a nutcracker.

3. Instrumental systems

The cracking of nutshells is intuitively an instrumental activity *par excellence*, and the nutcracker used to do it answers to common intuitions about what an instrument is. To see a system at work here, and tell what its components are, is less intuitively obvious, however. By introducing the idea of a nutcracking system, as a kind of instrumental system, we aim to capture the purposeful activity of cracking a nutshell, and to do so we include in the system both the person who cracks and the nut that is to be cracked. More exactly, an instrumental system is a structured complex whose constituents are intentionally arranged in such a way as to transform a particular kind of input object into an output in a particular way. We regard the object to be transformed by the system as being itself a systemic component: an instrumental system is one in which a component undergoes a transformation from one state to another. It is evident that an individual needs to be inside a system, and insofar be a constituent, in order to be transformed. This is simply to say that in order to be cracked, a nut has to be fixed between the handles of the nutcracker, and in order to be transported by a car, you need to literally enter the car and remain inside until arriving at the designated destination. Once the transformation is completed, however, the transformed object does not remain a component of the system; the system has finalized its program, as it were, and comes to a stop.

To keep the exposition as simple as possible, we shall be black-boxing several notions, not least the notion of instrumental system itself. We have little to say about instrumental systems *per se*. We are satisfied with modeling them as *n*-tuples. The rule we are imposing is that an instrumental system is a three-component system. The components are slots for a *user*, an *instrument*, and an *object*. The incumbents of these three slots are held together in two different ways, in the following manner. A *causal* relation obtains between instrument and input object. For instance, the cracking that extends from a nutcracker to a whole nut is a wholly causal relation susceptible to no other constraints but the laws of nature. Likewise, from a causal point of view it makes no difference whether the passenger serving as the input object of the vehicle serving as the instrument of a taxi is animate or inanimate; what matters is the passenger's purely physical properties such as weight, etc. An *intentional* relation, on the other hand, obtains between user and instrument. Hence, user and input are not related directly; rather their contact is mediated through the instrument. This relation, as we understand it, is double-edged: the *purposeful action* that the user engages in when operating the instrument properly (as it was designed to by its creator) has both a physical and a psychological dimension. This is to say that when a nutcracking agent grabs a nutcracker with the intention of using it to crack a nut, the psychological dimension is at play, and when he operates the nutcracker by exercising force on the nut through squeezing the nutcracker's handle, the physical dimension is at play.

Let 'IT' signify the slot for a generic intentional relation capable of obtaining between a user and an instrument, where the intentional relation is aimed at transforming an input object that is related by a generic causal-coupling relation, denoted by 'CC', to the instrument. Then we define an instrumental system as having the following constituents related in the following manners in the following overall structure:

$$\textit{Instrumental system} =_{\text{def}} \mathrm{IT}\langle \textit{User}, \mathrm{CC}\langle \textit{Instrument}, \textit{Object}\rangle\rangle.$$

The underlying intuitive idea is that user and instrument fit like a hand in a glove. The user uses the instrument to obtain the sort of result that the instrument was designed to help obtain. Neither user nor instrument can obtain the result alone.

The intuitive way to approach the question of interaction between user and instrument is to ask what 'button' on the instrument the user needs to 'push' to get it to work. Sometimes there is literally a button to push, as when getting yourself a cup of coffee from a coffee-vending machine. But oftentimes the ‘button’ is a figurative one. The ‘button to push’ in the case of a nutcracking system whose instrument slot is filled by a nutcracker (and not your thumb and index finger, say) is its handle. The handle was designed to be grabbed and held by a human hand and is the point of entry of the interface between artifact and human. If you want to become the user of a designed nutcracking system, you enter the system by taking hold of the handle of a nutcracker. You operate the system by following its directions of use. In general, a system’s directions of use is a sequence of procedures or instructions.

When someone is the passenger of a taxi, the ‘button’ to push is no longer mechanical, hence tangible, but linguistic, hence social. It is institutionalized that what you do is tell the driver your designation. This sort of communication may be rather rudimentary, as when uttering “To the airport!” or showing him a slip of paper with the address written on it, but some measure of communication is called for. It is knowledge of this ‘button’ that enables us to enter a fare-based, human-operated system and get it to work.

On a formal note, a system is a transformer that takes something from one property (or pool of properties) to another property (or another pool of properties). The first property (properties) will be had at an earlier point in time than the second property (properties), which goes to show that there is a built-in *dynamics*. In fact, this transformation over time is pretty much the whole *raison d’être* of engineered systems. We create (design, manufacture, implement, etc.) them in order to avail ourselves of means to obtain properties that we consider desirable and which it would be either impossible or difficult or inconvenient to obtain in other ways.

We will present a more detailed analysis of a driving system, since this serves to demonstrate how complex systems emerge from the hierarchical linking of instrumental systems. In the case of a driving system, the input is whatever is being transported, the instrument is a car and the user is a driver:

$$\textit{Driving system} =_{\mathrm{def}} \mathrm{IT}\langle \textit{Driver}, \mathrm{CC}\langle \textit{Car}, \textit{Passenger/cargo}\rangle\rangle.$$

The driver uses the car to transport him- or herself and possibly also one or more passengers or some goods from one location to another. Note that if the sole point of driving the car happens to be to transport passengers or goods, it is accidental that the driver, and not only the passengers and goods, gets transported. For all the driver cares, he or she might just as well be remote-controlling the car from a fixed position. The general point is that a driving system is realized the minute somebody assumes the role of driver of a car. It makes no difference if, for instance, the driver spends hours driving around in circles or follows a random itinerary. What matters is solely *that* the car is being driven, and not *how* or *why*, as long as this is done intentionally. The rider ‘as long as this is done intentionally’ serves to rule out certain kinds of operating a car’s steer and pedals. Someone merely tampering with steer and pedals fails to qualify as a driver, even if this tampering were to cause the car to move in a manner physically indistinguishable from the purposeful activity of driving. Whatever the car does is not a physical realization of that person’s intentions, for he has none, as far as driving the car goes; he is just hitting buttons at random. In contrast, being held up by an open bridge or in a traffic jam still qualifies as driving: being in motion is neither sufficient nor necessary for an activity to qualify as driving a car.

In our modeling of instrumental systems, it is of the highest importance that different slots receive different names. Even if most cars are driven by people with the intention to change their own position, e.g. from home to work, it would cloud the vision of what a driving system is to represent such systems by IT⟨*Driver-passenger*, CC⟨*Car*, *Driver-passenger*⟩⟩. The nature of the system does not change when different persons realize the roles of driver and passenger, and even a single person may have competing interests in the role of driver and in the role of passenger. Role conflicts are an important factor in understanding the behavior of complex hybrid systems, and models of such systems should be able to represent any potential conflict of roles. Whenever there is coincidence, it is at the level of particulars and not at the level of roles (i.e., two roles, one occupant).

Operating with roles in the definitions therefore enables us to distinguish, at the conceptual level, between roles that tend to be co-occupied but need not be. A second purpose, however, is that it enables us to maintain, in particular, that a car with a driver waiting for a fellow passenger to arrive is no less a driving system for that. Only it is an *idle* one. Idleness is one of three states or modes that a given instrumental system may be in. One pair of states are mutually exclusive: either a system is *operational*, or it is *idle* (not both and not neither). If a system is idle, it is thereby in one of two mutually exclusive sub-states: either the system is *capable of being operational* (only it is ‘switched off’), or it is *malfunctioning* (it is ‘closed down’). That is, malfunction is an extreme form of idleness: a malfunctioning system is incapable of operating. The difference is between being contingently and necessarily idle.

The most frequent cause of an instrumental system slipping into a state of malfunctioning is when its instrument slot is occupied by a malfunctioning artifact. Here we understand a malfunctioning artifact to be an artifact of a particular functional kind that is incapacitated as far as functioning as an artifact of its functional kind. Thus, if nutcrackers are designed to function as nutcracking devices, then a malfunctioning nutcracker is one that is incapable of cracking nuts. Strictly speaking, a malfunctioning instrumental system cannot contain an occupied user slot. If your nutcracker malfunctions, your nutcracking system malfunctions. You can at most *attempt* to use the system, but your attempt is bound to meet with failure.

A malfunctioning artifact is a common, though not the only cause of instrumental-system malfunction. An instrumental system with a perfectly functioning instrument may nevertheless come to malfunction if something goes wrong in the coupling of user to instrument – e.g. when someone’s hands keep slipping from the handles of the nutcracker while attempting to crack a nut, or when someone who knows how to drive a car accidentally pushes the wrong pedals or pushes them in the wrong order – or the coupling of instrument to object – e.g. when a nut is accidentally launched from the cracker or when a car passenger goes berserk and makes it impossible for the driver to carry on.

4. Nested instrumental systems

We decomposed a nutcracking system into user, nutcracker and nut, and a driving system into driver, vehicle, and passengers and/or goods. This might seem to suggest that the hybridity of instrumental systems were ‘decomposable’: the user is a person, the instrument is a hardware device, and as far as the input object is concerned, only its character as a physical object seems to matter. To be transported, only a car passenger's physical presence in the vehicle seems to be required, allowing him or her to sleep through the entire ride. This would, however, be an impoverished view of the role of passenger. Someone can use a driving system to get herself from one place to another without doing any driving herself. Her hands are not on the steering wheel, her feet are not on the pedals; the driver's are. The passenger uses a system, consisting of a vehicle *and* its driver, as her instrument. This is the sort of driving system that we normally have in mind when we talk of a *taxi*. In order to arrive at where she wants to go, a passenger cannot be merely physically present in the car and be asleep from start to end; she must become a user, the taxi's *client*, by telling the driver her destination. Or at least someone must. The roles of passenger and client need not be occupied by the same individual. The client can also use a taxi to transport somebody or something else. For example, an opera diva who is taken from the airport to the hotel by cab does not specify the destination, nor does she pick up the bill. Her impresario does both, and he or she specifies where the diva is to go. Or a bike rider may pay a cab driver to transport her bike from one place to another while not riding along herself. In this case the bike fills the passenger/cargo slot, while the bike rider is the client, i.e., fills the role of user of the driving system. Furthermore, an institution, rather than an individual person, may both decide the destination and pay the fare, though it cannot figure as passenger. It is a trivial prerequisite for being a passenger, though not a client, that what fills the role be physical.

We will call any driving system of this sort, where the instrument used by the user is itself a simple driving system, an *assisted-driving system*. In such a system, the vehicle that does the transporting is not operated by the client of the whole system but by a driver who for this reason becomes a component of the instrument used by this client. We define:

$$\textit{Assisted-driving system} =_{\mathrm{def}} \mathrm{IT}\langle \textit{Client}, \mathrm{CC}\langle \mathrm{IC}\langle \textit{Driver}, \textit{Car}\rangle, \textit{Passenger/cargo}\rangle\rangle.$$

This is to say that the concept of an assisted-driving system is defined in terms of the concepts of client, driver, passenger/cargo, and car, together with a particular arrangement. This definition does not entail that every real-world taxi is going to have a passenger or a client at all times, or indeed at all. It does entail that every taxi has a *slot* earmarked for one or more passengers or pieces of cargo and a *slot* for a client. Whether the passenger/cargo slot of a particular taxi is, at a given time, occupied or vacant is an empirical matter and as such beyond the scope of the definition. The relation 'IC' in $\mathrm{IC}\langle \textit{Driver}, \textit{Car}\rangle$ is partly intentional, as is the relation 'IT' in $\mathrm{IT}\langle \textit{Driver}, \mathrm{CC}\langle \textit{Car}, \textit{Passenger/cargo}\rangle\rangle$. But the intention is not directed at the transformation of an object but at being instructed to operate the car. The person filling the driver slot may assume these instructions to be directed at the transformation of something, but such assumptions lie beyond the characterization of the $\mathrm{IC}\langle \textit{Driver}, \textit{Car}\rangle$ system.

The conceptual transition from *driving system* to *assisted-driving system* is a complicated one. Whereas the former involves transactions between an artifact and a human, the latter involves transactions between two humans fulfilling two sets of different roles. If we suppose the roles of client and passenger to be filled by the same person, as is customary, then from the passenger's perspective a taxi is an assisted-driving system that is operated by communicating with the driver. The passenger's motive for inserting him- or herself into this system is to undergo a transformation: from being where you are to going where you want to go.

The perspective between passenger and driver can, however, be *inverted* to form another layered system. From the perspective of the driver, more precisely, from the perspective of the person who fulfils the role of driver, a taxi is a means to generate income When the table is turned, the taxi becomes part of an *income-generating system*, and the ‘button’ the driver ‘pushes’ is to let the passenger know what the fare is. We define:

$$\textit{Income-generating-by-assisted-driving system} =_{\text{def}}$$
$$\mathrm{IT}\langle \textit{Income-generator}, \mathrm{CC}\langle \mathrm{IT}'\langle \textit{Client}, \mathrm{CC}'\langle \mathrm{IC}\langle \textit{Driver}, \textit{Car}\rangle, \textit{Passenger/cargo}\rangle\rangle, \textit{Beneficiary}\rangle\rangle.$$

Now the driver uses the entire system IT′⟨*Client*, CC′⟨IC⟨*Driver*, *Car*⟩, *Passenger/cargo*⟩⟩, an assisted-driving system, as an instrument to earn a living, which in dynamic terms comes down to increasing his income over time. When the driver earns his or her own income, he or she fulfils three different slots of this system – income-generator, beneficiary, and driver –, but these slots represent different roles, meaning that they could be fulfilled by different persons. The driver could give the money he or she earns to another person or persons, and if he or she falls ill, someone else could stand in (‘drive in’) for him or her.

The income-generating-by-assisted-driving system is an example of a system with three levels. It furnishes an example of how our approach allows the conception of systems with an arbitrary number of levels. In the final section we sketch the rules to form such systems out of the basic constituent kinds *user*, *instrument* and *object*.

5. Taxonomy of complex instrumental systems

In the previous section, we have presented two examples of systems where one of the slots, the instrument slot in both cases, is filled by a particular that is itself systemic. In the income-generating-by-assisted-driving system, the instrument is a complete instrumental system, an assisted-driving system, with particulars filling the user, instrument and object slots. The instrument slot of the assisted-driving system itself is also systemic, but this system is not an instrumental system, since it lacks an object slot.

These examples raise the question in how many other ways the slots of an instrumental system can be filled by systems. Without being able to argue this extensively here, for lack of space, we hold that the object slot of any instrumental system can also be filled by either an instrumental system IT⟨*User*, CC⟨*Instrument*, *Object*⟩⟩ or a smaller system of the form IC⟨*User*, *Instrument*⟩. Additionally, both the instrument slot and the object slot of any instrumental system can be filled by things with the system structure CT⟨*Instrument*, *Object*⟩. This is also not an instrumental system, since it lacks a user. It is a system where one part is transformed through being causally coupled to another part, but this transformation is not initiated intentionally by a user who ‘presses a button’; the causal coupling between instrument and object is sufficient for initiating the transformation.

Instrumental systems with the instrument or the object slot filled by an instrumental system correspond to two important *types* of instrumental systems. Let us call an instrumental system aimed at transforming its object by the activity of ϕ-ing a ϕ-ing system, which has the structure IT⟨*ϕ-User*, CC⟨*ϕ-ing Instrument*, *Object*⟩⟩. With this system in the instrument slot we get a:

Benefiting-from-ϕ-ing system $=_{\text{def}}$
IT⟨*System-user*, CC⟨IT′⟨*ϕ-User*, CC′⟨*ϕ-ing Instrument*, *Object*⟩⟩, *Beneficiary*⟩⟩.

The Income-generating-by-assisted-driving system we have introduced in the previous section is an example of such a system. With a ϕ-ing system in the object slot, on the other hand, we get a:

Φ-ing-system-regulating system $=_{\text{def}}$
IT⟨*Regulator*, CC⟨*Instrument*, IT′⟨*ϕ-User*, CC′⟨*ϕ-ing Instrument*, *Object*⟩⟩⟩⟩.

A ϕ-ing-system-regulating system regulates the activity of ϕ-ing. An example is when speed bumps or traffic lights are used to regulate the driving activity of the driving systems that pass across or through them.

The three systems that can fill the instrument and object slots of instrumental systems have a process character; they correspond to an ongoing activity. Therefore none of them can fill the user slot of an instrumental system. The user slot must be filled by a particular capable of intentional action, and the processes corresponding to the three systemic components introduced above do not qualify for this. Instead of being present as processes, however, a slot can also be filled by a particular with a systemic character that is due to its having undergone a transformation by an (instrumental) system. We indicate the result of a systemic transformation by ‘APP’, short for ‘applied’. This enables us to distinguish between the two results APP(IT⟨*User*, CC⟨*Instrument*, *Object*⟩⟩) and APP(CT⟨*Instrument*, *Object*⟩): the transformation undergone by the object is intentionally brought about by the user in the former but not the latter. Since the system IC⟨*User*, *Instrument*⟩ does not contain a slot for an object to be transformed, nothing can correspond to APP(IC⟨*User*, *Instrument*⟩). We hold that the remaining two product types APP(IT⟨*User*, CC⟨*Instrument*, *Object*⟩⟩) and APP(CT⟨*Instrument*, *Object*⟩) can fill either the user slot or the instrument slot or the object slot of any instrumental system. Whether for any kind of system realizations exist in which things filling the three basic slots have any of the above structures is an empirical matter. For a particular system, this would amount to opening the black boxes that fill these slots. We see no *a priori* arguments why it would be impossible to design a system of a particular structure for as many slots as we may wish to further analyze.

By way of summary, given our starting point of the three different slots of user, instrument and input object, we restrict the stratification of instrumental systems through the nesting of components only in three ways: (1) no system can consist merely of slots for a user and an object, since nothing would then mediate the interaction between user and object; (2) a user slot cannot be filled by a process-type system; (3) a system consisting only of the coupling of a user and an instrument cannot result in a transformed object. These restrictions leave room for twelve different rules of logically possible substitution specifying how slots in an instrumental system can be filled by systemic constituents that again contain slots for a user, an instrument and/or an object. These recursive rules furnish the means to model systems with stratified layers of arbitrary complexity, where physico-intentional hybridity can be present at any level. We see this approach as a promising tool for the modeling of large-scale complex socio-technical systems.

Acknowledgements

The research reported in this paper was supported by the Next Generation Infrastructures Foundation (http://www.nginfra.nl/).

References

[1] M. Franssen and P. Kroes, “Sociotechnical systems,” *A Companion to the Philosophy of Technology*, ed. by J. K. Berg Olsen, S.A. Pedersen and V.F. Hendricks, Wiley-Blackwell, Chichester etc., 2009, pp. 223-226.

[2] M. Ottens, M. Franssen, P. Kroes and I. van de Poel, “Modelling Infrastructures as Socio-technical Systems,” *International Journal of Critical Infrastructures*, Vol. 2, No. 2/3, 2006, pp. 133-145.

[3] B. Wilson, *Systems: Concepts, Methodologies, and Applications*, 2nd edition, John Wiley, Chichester etc., 1990.

[4] V. Hubka and W.E. Eder, *Theory of Technical Systems: A Total Concept Theory for Engineering Design*, Springer-Verlag, Berlin etc., 1988.

[5] T.P. Hughes, “The Evolution of Large Technological Systems,” *The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology,* ed. by W.E. Bijker, T.P. Hughes and T. Pinch, MIT Press, Cambridge, MA and London, 1987, pp. 51-82.

[6] M. Bunge, *Treatise on Basic Philosophy, Volume 4, Ontology II: A World of Systems*, Reidel, Dordrecht etc., 1979.