. . . THE ABILITY OF 'MATTER' TO ENABLE A BRAIN WITH 'CONSCIOUSNESS', WHICH CAN CONSTRUCT A 'THEORY OF THE WHOLE UNIVERSE' IS AN OUTSTANDING PHENOMENON. BUT BY PRINCIPAL REASONS IT IS NOT POSSIBLE TO WRITE A 'COMPLETE' THEORY (GOEDEL 1931, HAWKING 2002 [1]). INCLUDING THE 'GENERATOR' OF THE THEORY, THE THEORY-GENERATING BRAIN, TURNS ENGINEERING INTO PHILOSOPHY . . . THIS IS THE DISTRIBUTED ACTOR-ACTOR INTERACTION (DAAI) PARADIGM . . .

GERD DOEBEN-HENISCH

[1] Kurt Goedel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, i. *Monatshefte fuer Mathematik und Physik*, 38:173–98, 1931; and Stephen Hawking. Gödel and the end of physics, 2002. http://www.hawking.org.uk/godel-and-the-end-of-physics.html . Another URL with the audio file: http://www.damtp.cam.ac.uk/events/strings02

GERD DOEBEN-HENISCH

DISTRIBUTED ACTOR ACTOR INTERACTION [DAAI]

JANUARY 9, 2020 - VERSION 15.06.04

UFFMM.ORG

PUBLISHED BY UFFMM.ORG

UFFMM.ORG, ISSN 2567-6458

First printing, May 2019

Contents

List of Figures

Dedicated to those who gave us the prior experience and the inspiring ideas to develop the view offered in this book..

Preface

An DAAI Course Program: This text presents a short, condensed version of an analysis using the DAAI (Distributed Actor-Actor Interaction) paradigm, which can be handled within one semester term of a master program. But even this short version tries to bring together such diverse topics like *Human-Machine Interaction (HMI)*, *Systems Engineering (SE)*, *Artificial Intelligence (AI)*, *Cognitive Science (CogS)* and *Philosophy of Science (PhS)* in one coherent framework. This text is intended to introduce a complete process from starting with a problem, analyze the problem in an DAAI manner, test the result and stop.

Web Site This small text is located as one sub-topic at the main website https://www.uffmm.org/2019/05/12/aci-frontpage/.

Terminology: HCI - HMI - AAI - DAAI From the history of computer after the World War II[2] one can see that the development of the computer hardware induced steadily new ways of usages of computers, which simultaneously induced new requirements for the professional users of a computer. In the early beginnings it was a challenge to have the right programming languages for coding ideas and to enable more human like interfaces. This was the age of *HCI (Human Computer Interaction)*. The then occurring spreading of computer technology in more and more areas of everyday working environments induced a change from interactions with typical computers only to interactions with technical environments in general, where the computer is now an embedded technology, hided in the environment. This was the age of HMI (Human Machine Interaction). The further development of *Artificial Intelligence (AI)*, especially in its diminished format of *Machine Learning (ML)*, transformed the *classical* machine concept into a new, *smart* machine concept, which turned the boundaries between man and machines into a fuzzy matter, where the concept of an actor can now mean some robot, some smart program as well as a human person. This is the age of *AAI (Actor-Actor Interaction)*. Somehow simultaneously to the dawn of the smart machines we have the advent of *distributed working places* interacting in a pseudo entangled way: Although the different workers at different locations around the world are spatially separated they are connected by nearly real-time data streams. This turns the locally separated work-places into an entangled work-place appearing as *one* data-enabled place.This characterizes a *Distributed AAI* paradigm. The analogy to entangled states in Quantum Mechanics is striking.[3]

[2] For a first introduction see the two human-computer interaction handbooks from 2003 and 2008, and here especially the first chapters dealing explicitly with the history of HCI (cf. Richard W.Pew (2003) , which is citing several papers and books with additional historical investigations (cf. p.2), and Jonathan Grudin (2008) . Another source is the 'HCI Bibliography: Human-Computer Interaction Resources' (see: http://www.hcibib.org/), which has a rich historical section too (see: http://www.hcibib.org/hci-sites/history).
Richard W. Pew. Introduction. Evolution of human-computer interaction: From memex to bluetooth and beyond. In J.A. Jacko and A. Sears, editors, *The Human-Computer Interaction Handbook. Fundamentals, Evolving Technologies, and emerging Applications*. 1 edition, 2003; and Jonathan Grudin. A Moving Target: The Evolution of HCI. In A. Sears and J.A. Jacko, editors, *The Human-Computer Interaction Handbook. Fundamentals, Evolving Technologies, and emerging Applications*. 2 edition, 2008

[3] See the nice article 'entangled states' in Wikipedia at https://en.wikipedia.org/wiki/Quantum_entanglement.

1
The 'All in One View'

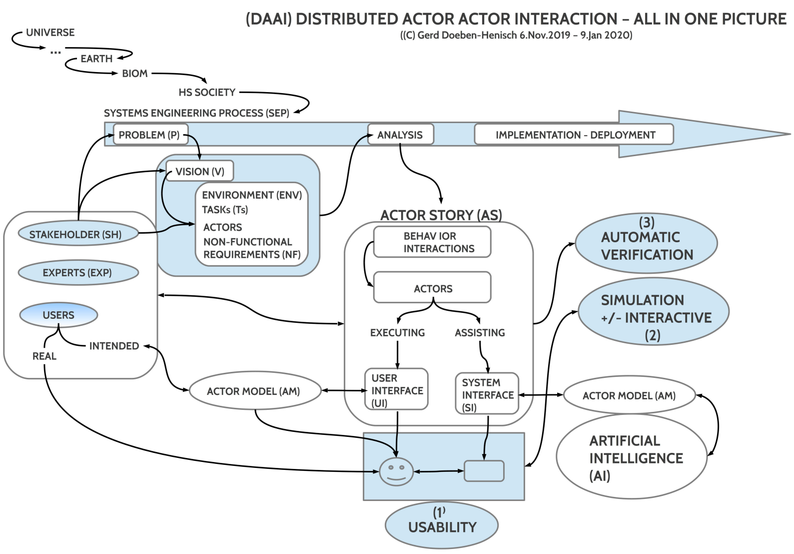

Figure 1.1: DAAI analysis, the 'All in One View'

The figure 1.1 shows in one view all the topics which will be covered in the DAAI paradigm as proposed in this text.

FIND A SOLUTION: The whole machinery of the *Distributed Actor-Actor Interaction Analysis* – short: DAAI analysis – is rooted in the idea to find an *optimal solution* for a *given problem*. This solution has to be given as a physical something which mimics the intended *interface* of a technical system in a way, that a *real user* can *test* the interface by trying to *solve a given task* in a *given environment*. To qualify an interface as *optimal* requires some *objective benchmarking* in a way, which everybody can accept and repeat. This kind of benchmarking is usually called *usability test* and it is nothing else then a special kind of *measurement*. In the usability test someone *compares* an X to be measured with an Y which serves as an accepted *norm*, as an accepted *standard*.[1] During an usability test a real user is interacting with a real something of an intended interface of a technical system. The primary subject for the measurement is given by this *sequence of interactions* which represents the *behavior* of the user as well as of the interface. But what are the standards for comparison?

[1] The international accepted measurement standards are managed by the *BIPM: Bureau International des Poids et Mesures* which is associated with many member states (see URL: `https://www.bipm.org/en/about-us/`)

ANNOUNCEMENT OF A VISION: The primary standard is given by that *vision*, which a stakeholder – which can be a large group – has announced as the answer to a *problem*, which he hast stated before. The vision has to include certain *tasks* which should be possible to be done by certain *actors* in a certain *environment*, further characterized by some *non-functional requirements (NFRs)*. Such non-functional requirements are calling for general properties like 'being save', 'working in real-time', 'being competitive in a certain market', and the like.

ELABORATE THE VISION: The vision is a first *sketch*, a first *outline*, a very *broad direction* where to go, but it is not yet clear enough for an exact specification. This has to be done from a group of *experts* which have enough experience, knowledge, and communication skills to *translate* the vision gradually into a more concrete description, such that the description worked out within an DAAI analysis finally can be used as that *standard* needed for the usability test. This more concrete specification is in the following text called *actor story (AS)*, whereby the actor story can be extended by *actor models (AMs)*.

ACTOR STORY (AS): An actor story has to be realized as a collection of *basic facts* where each basic fact can be decided as *being true* or being *not true* or judged as being *not decidable* with regard to the before selected environment. Such facts have to be organized as sets of facts where one set represents a *state*.[2] With regard to states one has to assume *basic functional units* which describe basic transformations between two consecutive states S and S': By *deletion* a fact from S will not occur anymore in S'. By *creation* a fact F not yet in S will occur in S'. There can be more than one functional unit operating on a state S to transform S into a consecutive state S'. A sequence of states and transformations of the states defined by functional units is called here an *actor story (AS)*. The functional units can be interpreted as *interactions* caused by *actors* which are part of a state. The set of all interactions represents the *behavior* of the actors.

[2] often also called *situation*, *scenario* or *scene*.

ACTORS - ASSISTING AND EXECUTING: While in the past the distinction between the interface of the system and the human user has been the predominant view, it makes today more and more sense to talk of *actors* with the new distinction between the *assisting* actor – the classical interface of the technical system – and the *executing actor* – the classical human user. Using different *functional roles* one can view these roles as *slots* which can be *filled* with different kinds of real entities as long as they can provide the functionality which the role requires.

ACTOR MODELS (AM): The new formal rigor in the description of the actor story allows a new enhancement of the actor story by introducing *additionally* so called *actor models (AM)*. While an actor story provides only a *3rd-person view* of the participating actors by describing their *observable behavior* it can be helpful or even necessary to be able to describe the *internal functionality* of an actor to enable some *additional rationality* in the understanding of the processes. The interaction between the actor story and the participating actor models is determined by the individual interface of an actor: everything the actor story states about the behavior of an actor in a certain situation has to be provided by the internal functionality of the actor model. But as soon as the behavior of an actor will be *determined by its internal functionality* this can induce a *surplus* of *possible behavior* compared to that behavior which is specified by the actor story. In case of *deterministic* actors this can be managed in most cases, but with *truly learning actors*[3] the generated behavior can surpass that behavior which is specified in the actor story. This transforms the specifications of the actor story into a somewhat *fuzzy* space of possible events.

[3] this is at least the case with human actors!

BIOLOGICAL - NON-BIOLOGICAL:
The primary reference for the modeling of the internal functionality of an actor is given by the actor story which follows the vision of the stakeholder. There is no specific need for a certain type of modeling as long as the primary reference will be matched. In case of human actors it can be of help to follow the empirical structures of biological systems in the modeling of the internal functionality of the actor if it is important to match the behavior of real persons as close as possible. But even if this claim is an issue it is not completely defined what kind of a formal model will serve this requirement best. This ambiguity results from the fact that the *behavior based* sciences, the *physiology (including the brain) based* sciences, as well as the *phenomenological* sciences are not yet unified today. These three views coexist one besides the other and it is not clear when and how a more fruitful integration will happen in the future.

(ADAPTIVE) COMPLEX SYSTEMS ((A)CS): Following the basic intuition, that in a complex system one element depends from the behavior of another element[4], every state of an actor story is a complex system. Weaver (1958) [5] calls this kind of complexity *organized* complexity which he distinguishes from *disorganized* complexity, where the elements of a system are not really dependent from each other. Because *human* actors are not only complex systems, but even *adaptive complex* systems, one has to state that every state of an actor story can represent a truly social system, where the different individual actors are depending in their behavior from the behavior from the others, and they can *change* their behavior.

[4] cf. Miller & Page (2007)
John H. Miller and Scott E. Page. *Complex Adaptive Systems. Introduction to Computational Models of Social Life*. Princeton University Press, Princeton - Oxford, 1 edition, 2007

[5] Warren Weaver. A quarter century in the natural sciences. In *Rockefeller Foundation Annual Report, 1958*, pages 7 – 15. Rockefeller Foundation, 1958. URL: https://assets.rockefellerfoundation.org/app/uploads/2015053012 Report-1958.pdf

EMERGENCE: Looking to the behavior of sets of systems it is well known that the behavior of the individual systems can be different from the behavior of the whole set of systems. This phenomenon is called *emergence*.[6] But following the distinction between a disorganized and an organized complexity one can also distinguish between a *weak* and a *strong* emergence.[7] In case of an actor story we have a *strong emergence* because the overall behavior of a state depends on the relationships between its actors in this state. Systems with a strong emergence are not reducible to more simpler elements!

[6] cf. Miller & Page (2007)

[7] cf. Doeben-Henisch (2020) `https://www.uffmm.org/wp-content/uploads/2019/06/review-santa-fe-8jan20-miller-page-2007-cc1-4.pdf`

ARTIFICIAL INTELLIGENCE (AI): Today the *mainstream* induces the impression that *smart* machines are already there and that these will in the future improve steadily until a point, where the homo sapiens[8](cf. Krause et.al. (2019) [9]) seems to be without a further point. This text here will advocate the stance that this opinion is completely wrong. The property of a machine of being more fast and simultaneously of being able to process more and more data is impressive, but does not touch any of the big problems which have to be solved today and in the near future. Nevertheless with the explicit introduction of actor models in the DAAI paradigm one can *include* all the nice topics of *artificial intelligence (including machine learning)* into the actor models. The actor story is then a formally defined environment for the behavior of the introduced smart actors. The instrument of the actor story allows therefore the *integration* of *human and non-human actors* with artificial intelligence in one coherent framework.

[8] 'home sapiens' is the branding for that kind of life form which appeared in Africa about 600.000 years ago, and which has spread since about 50.000 years ago from Africa throughout the world. We all are descendants from them.

[9] Thomas Krause, Johannes; Trappe. *Die Reise unserer Gene: Eine Geschichte über uns und unsere Vorfahren.* Ullstein Buchverlag, Berlin, 5th edition, 2019

(INTERACTIVE) SIMULATION (IS): An actor story as such is already a *dynamic* concept dealing with transformations of states by applying functional units. Mathematically an actor story is a *graph* which can be interpreted as the *execution graph of an automaton*. If one takes this implicitly defined automaton as a *simulator* one can easily define an actor story as a *simulation*.This allows a better understanding of the space of possible states, especially in complex cases. To turn a normal simulation into an *interactive* one is straightforward. This opens new applications to use an actor story also for *training* and *learning*.

AUTOMATIC VERIFICATION (AV): If one takes the actor story as a graph one can use it within an *automatic verification setting* too.[10] This allows the analysis of very big and complex cases in a purely automatic and fast way. While normal simulations can reach quickly the timely limits of the performance of human users, an automatic verification can work without a human person interrupting the process and can search the complete search space for a given level of computation to find *all* possible answers. This feature – here called the *Greek oracle function (GOF)* – can probably become the most important feature for all practical applications .

[10] See e.g. Baier and Katoen (2008)
Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking.* MIT Press, Cambridge (MA), 1th edition, 2008

BENCHMARKING REFERENCES: The actor story in combination with simulation and automatic verification can be used as a benchmark in more than one way.

1. For the objective of *usability* the actor story as standard specifies which *tasks* have to be done in a certain environment by which users in a concrete decidable way. A test can show the *percentage* of the tasks which will be fulfilled (a measure for completeness); the *number of deviations* which occur (a measure for errors); the *learnability* of the tasks by the *test persons* using repetitions[11]; and the *user satisfaction* after completion of a test run.
2. The *stakeholder satisfaction* with regard to his vision can be measured (i) in interaction with a *simulated actor story* where the perception and the dynamic of the actor story can match the vision with full experience, as well (ii) by the results of *automatic verification* testing the *non-functional requirements* in all possible configurations within a given time window.

[11] Based on the change of completion and errors within a time window.

3. What can not be tested by an actor story that is the success in the market. This success is depending from many additional factors which are beyond the full control of the offering company.

IMPLEMENTATION: The next phase in the systems engineering process after the DAAI analysis is the *logical design phase* to prepare the *implementation phase*. The input for these two consecutive phases is given by the requirements for the expected behavior of the system. Having a complete actor story at hand one has all specifications which are necessary. In case of actor models one has an extension of this specification because the internal functionalities of the actor models realize at least the format of a logical specification like those needed in the logical design phase or – depending from the overall framework – the internal functionalities of the actors are already part of the final implementation.

2
Problem and Vision Statement

CONTEXT: In chapter 1 the *vision* statement is mentioned as the answer to a *problem* statement, which the stakeholder has announced before. The vision statement functions as the main point of reference for benchmarking the actor story with possible actor models worked out by experts to find a solution to the problem in the light of the vision. What can be said about both statements?

THE PROBLEM STATEMENT: To enable a vision one needs a point of reference to a situation which has been classified as a *problem*. The meaning of the word 'problem' depends highly from the stakeholder's view of the world. This view can be associated with rather *objective* facts, but can depend too from more subjective preferences or 'intuitions' which can not be completely 'explained' by known reasons. All really innovative products or services have in the beginnings a certain amount of *vagueness* and *hope* for the market success and the usability of new features. Thus it will need a longer process revealing different kinds of evidences to support the lacking rationality in the beginning. Therefore to classify a situation as being a 'problem' depends from the availability of a world view which sees some opportunities in the future. Thus to classify a situation as a *problem* you need some minimal vision of improvements, and to state something as a *vision* you need a given situation as point of comparison to illustrate the *different new approach*.

THE VISION STATEMENT: What is needed to be able to depart from a given situation which looks as *less promising* then *something new*? With regard to a certain market and the production/ deployment process there exist some rather objective criteria which have to be met to be 'successful', but to 'evaluate' the vision in the light of such rather objective criteria one has to have *sufficient knowledge* about the *content* of the vision. Minimal factors for such a knowledge are (i) the kinds of *tasks (T)* which should be possible with the new product/ service, (ii) the kinds of *actors (A)* which will be involved in the realization of the tasks using the new product/ services, (iii) the intended *environment (ENV)* in which the tasks shall be realized, and usually (iv) some *non-functional requirements (NFRs)* characterizing the product/ service not only in one situation but in *all situations* associated with the realization of the tasks. The vision text should be explicit enough that one get a sufficient 'idea' of what could be meant, but at the same time it should not be to detailed to allow that the experts can bring in a *maximum of innovative ideas* to work out an exciting new product or service.

3
Actor Story and Meaning

CONTEXT: In chapter 1 the actor story (AS) is the connection between the vision as the starting point and the main benchmark on one side and the different kinds of evidences to confirm the actor story as a possible realization of this vision. The evidences emerge primarily from the working experts trying to translate the vision in real processes. Further evidences can be gained through usability tests, through simulations, as well through automatic verification processes.

ACTOR STORY AND THE VISION STATEMENT: The vision statement from the beginning (cf. chapter 2) tells something about the intended tasks which should be realized, the environment, where this should happen, the actors which should be the intended players in the field, as well some non-functional requirements, which have to be acknowledged for this vision. In the vision statement this is only communicated very roughly, as a sketch, an outline; there are no assumptions about the details where and how this should be realized. These details have to be filled in to give the needed detailed picture which can enable a technological solution which finally hopefully can work successfully. This is the job of the actor story: tell the real story with enough details and with a clear logical order.

LANGUAGE AND MEANING: It is the job of the experts to construct an actor story which satisfies all expectations raised in the vision statement. This requires from the experts to use a *language* for communication. The primary approach is to start with the most common language, the *everyday language* which can be assumed to be understood by everybody or – if the experts are representing a multicultural group – to use that language which is most common for all. A main characteristic of an everyday language is that phenomenon which usually is called *meaning*. If someone utters a language expression like 'There is a red car' then usually everybody who is hearing this utterance will be stimulated by this utterance to look for some object in his environment which 'looks like a care which is red'. Thus the language expression as such is different from that something to which it is pointing to. Moreover, the language expression as such, the sound which one can hear, is also not itself the pointer![1] The pointer from language expression to something else is located in our brain.

All children have to *learn* step by step which kinds of language expres-

[1] If somebody would utter a German expression like 'Da ist ein rotes Auto' instead of an English expression, then all those who cannot understand German would not react.

sions have to be *associated* with which kinds of other things. For this they have to collect more and more *pointers in their head* which allow such a switching from language expression to something else and from something (the read car there before you) to a language expression.[2] The set of all these pointers together constitutes the *meaning function* μ of a language L which is mapping from the *expressions* of a language L_{expr} to the *meaning* of the language L_{mean} and vice versa. The meaning function represents therefore a bidirectional mapping $\mu : L_{expr} \leftrightarrow L_{mean}$. It is known from *developmental psychology* that the children not only have to learn the meaning function μ but also the *structure of the world of objects* (cf. Harris (1992)[3], Fletcher and MacWhinney (1995)[4], and Bloom (2000)[5]). This is the reason why children will start playing with words and language expressions only if they have built up a sufficiently rich *structure of objects* in their heads which can serve as the counterpart of the possible language expressions to enable the meaning of language. And they have to learn with the meaning function of a language the *structure of the language expressions* too.

[2] In extreme cases the 'other' to which a language expression is pointing can be another language expression! This happens when we are talking about our talking or writing.

[3] Margaret Harris, editor. *Language Experience and Language Development: From Input to Uptake*. Lawrence Erlbaum Associates, Hillsdale (USA), 1992

[4] Paul Fletcher and Brian MacWhinney, editors. *The Handbook of Child Language*. Basil Blackwell Ltd, Oxford (UK), 1995

[5] Paul Bloom. *How Children Learn the Meanings of Words*. The MIT Press, Cambridge (MA), 2000

COGNITIVE SYSTEMS: From this follows the general assumption, that there are at least three systems which have to be learned: as the primary system the *structure of objects*, and as secondary systems the *language expressions* together with the *meaning function*. Because these learning processes are different in every person there is *no exact 1-to-1 congruence* between the different individual meaning functions; they *always differ* and make the learning and the usage of natural languages an *enduring adventure*. This is the reason for the well known *semantic gap* which is a steady source of misunderstandings and errors caused by this gap (cf. Doeben-Henisch and Wagner (2007)[6]).

[6] Gerd Doeben-Henisch and Matthias Wagner. Validation within Safety Critical Systems Engineering from a Computational Semiotics Point of View. In *Proceedings of the IEEE Africon2007 Conference*. IEEE, 2007

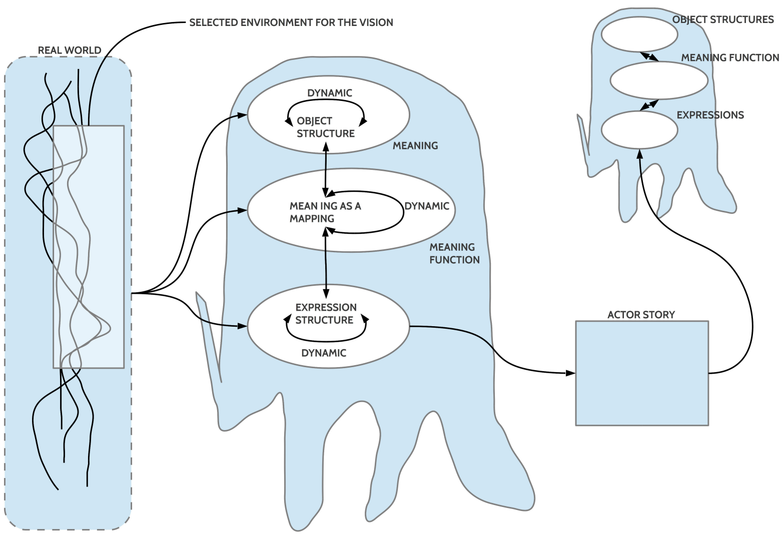

Figure 3.1: Minimal assumptions about the interacting language related systems in the head of the participants

INDIVIDUAL MEANING FUNCTION: Knowing about this structure of a language and its way of encoding reality in a dynamic fashion located in the brain one has to look to an actor story as a *symbolic space* realized by language expressions whose meaning function has to be presupposed in the heads of the participating experts and stakeholders. Whatever some expert will write in an actor story he is using his actual encoded meaning and he has to presuppose that all the other experts will interpret the expressions of the actor story in a *sufficiently similar way*. In everyday communication the different experts can always make small tests whether this assumption of the *sufficient same meaning* is *true* by checking how the other expert is responding to the text and how the other expert will associate the text with certain kinds of objects or behaviors in the shared real world. The expressions of the everyday language as such show no meaning; you really have to *know* the meaning function to reach the *intended* meaning.

PICTORIAL MODE, COMICS: If you would instead of the normal everyday language use the kind of language known from *comics* then the situation is changing. Comics are using a mixture of drawings and simple language expressions. The drawings depict objects and events from the real world directly without the encoding of language expressions. Thus publishing comics in different countries with different languages usually works without changing the drawings. Because the real-world experience based on our visual and acoustic senses is causing somehow the *same* structures in our heads which can be represented by certain kinds of drawings on can use the same drawings for different spoken and written languages. The minimal language expressions in comics are mimicking the occurrence of language in the real world. One can *see* a person as part of a scene, but when this persons starts *talking* it would become difficult in a drawing to represent the talking again as a sequence of drawings. Thus a comic-like style of communication can improve the understandability of a symbolic structure a lot. In this texts this style of communication is called *pictorial* mode differentiated from the *textual* mode.

Summing up: the basic principle of the *pictorial* mode of communication is the assumption, that there is a *sufficient similarity* between the *real drawings* and the *learned object structures* of the real world. Because the learning history of two experts can differ there exists no objective criterion whether a drawing is *sufficiently similar*. While one person needs only a *few hints* to identify the intended meaning by some drawing, another person needs possibly *more hints* or will not be able to identify the meaning at all because he has never seen the intended object before.

For the announced possibility of *simulation* and *automatic verification* the question arises, how one can *translate* a pictorial and an everyday textual mode into a *mathematical mode*, which can be processed by a computer? How one can translate the natural meaning function in a way that a formal symbolic space will work?

THE ACTOR STORY UNIVERSE as used in this text is a structure trying to match the natural cognitive structure of human persons as well as to match the requirements for a formal processing of actor stories. The figure 3.2 can look at a first glance a bit 'quirky' but the basic idea is very simple.

PICTORIAL AND EVERYDAY TEXTUAL: The *primary assumption* is given with the basic cognitive systems for object structures, expression structures,

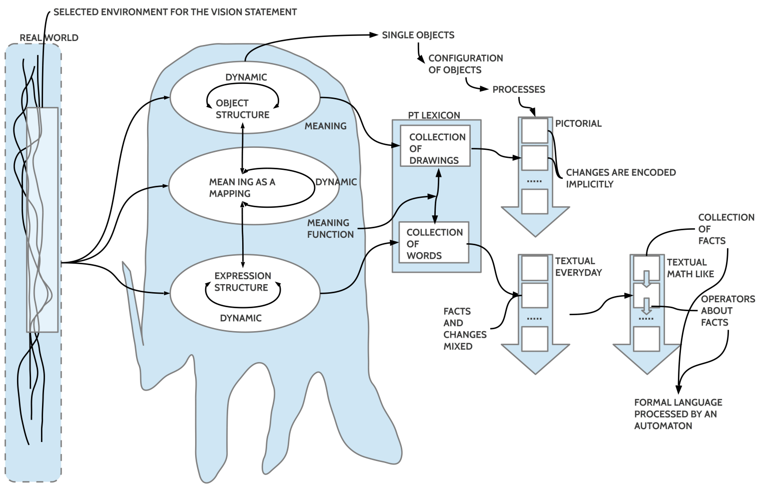

Figure 3.2: Different modes of actor stories with regard to meaning

and the meaning function matching objects and expressions. These systems are *inside* our heads somewhere in the brain and cannot be shown to others. To support a more transparent working one can *represent (encode)* parts of the object structure in a collection of *visual drawings* as well as parts of the expressions in a collection of *readable words*. Presupposing the known meaning function one can *construct an implicit mapping* between visual objects and written words by compiling a *lexicon* associating viewable objects and readable expressions, a *pictorial-textual lexicon (PT-Lexicon)*. Grounded in such a lexicon one can represent real-world processes either with a sequence of drawings (like in a comic or story board) or as a sequence of language expressions (like in a story). The highest understandability can be reached by combining the drawings with the written expressions. This would result in a *picture story* which has to be distinguished from a *comic*. In a picture story you have the pictorial and the textual mode which are each in principle 'self-contained', independent from each other. In a comic the text is complementing the drawings, but the text as such is not self-contained.

While the *static* objects, properties and relations can be represented directly by drawings or by written expressions, in cases of *changes* this is different.

Drawings can represent changes only implicitly by successive pictures which are *different*. While this difference is perceivable the change as process is not. Real persons can have a *change experience* on account of our brain which cuts the stream of perception into small time-slices which can be stored in some way and processed and this stuff can constitute the counterpart in a meaning function, but this *internal* meaning can not be shown as such. Perceiving a drawing before the change and a drawing after the change then this perception can trigger the activation of a *similar internally encoded change experience* linking the two drawings, but this change experience as such can not be drawn.

In the case of *written expressions* the situation is different compared to drawings. Because written expressions depend completely from some meaning function they never point directly to some meaning. Therefore whether an expression talks about static matters or about changes makes no difference in the expressions. Thus an expert can utter "There is a red car" as some static affair, but he can also utter "The read care is moving". While the accpompanying perception of a real process generates some internal meaning for the expression 'is moving', the expression as such gives no hint what kind of perception this is.

Thus in *textual* mode one can use expressions for changes without showing what kind of meaning these changes imply, in *pictorial* mode one can not draw changes as such too, but one can show the *difference* which emerges by the change.

MATHEMATICAL MODE: How do these two different aspects of reality fit to an actor story in *mathematical* mode which is needed for simulation and automatic verification?

The mathematical representation used in this text is also a collection of language expressions but with two different usages. One kind of mathematical expressions is used as *fact expressions* describing the *static* configuration of objects constituting a situation, also called a *state*. The other kind of mathematical expressions is used to describe the *differences* between two consecutive states. These kinds of expressions are called *change expressions*. Thus if in state S one fact expression F is occurring but not in the consecutive state S' then the consecutive state is classified as being *different* compared to the preceding state S. Similarly, if in the consecutive state S' a fact F' is occurring which has not yet been part of the preceding state S then state S' is called to be *different* too. These change expressions do not talk about objects, properties or relations between them but they are talking only about whole *states* and their *fact expressions*. As in the case of the pictorial or the everyday textual mode the intended meaning of changes recognizable by the occurring differences is in the case of the mathematical mode also grounded in the internally encoded meaning of changes. To judge whether the difference between two sets of facts of two consecutive states is representing a real change or not will still depend from the available meaning functions. But as soon as a mathematical encoding of facts and changes is realized these formal expressions will constitute a *formal language* which can be *processed by an automaton*. This enables the usage of an appropriate *computer*. With this possibility *simulation* and *automatic verification* can be realized.

4
Actor Story Modes: Textual, Pictorial

CONTEXT: In the preceding chapter a distinction between the *pictorial* and the *textual* mode of an actor story has been introduced, whereby the textual mode has been distinguished further by a textual mode with *everyday language* and a *mathematical language*. These distinctions will be illustrated in this chapter.

4.1 Everyday Textual Mode Example

In this section a short example for a simple actor story in *everyday textual mode* will be given. The starting point is a short *vision statement*. The vision statement is an answer to an assumed preceding *problem statement*, where it has been criticized that the work room of some workers is not secure enough, because everybody can enter the room.

VISION STATEMENT: The working room of a worker will be made secure by installing a door with an electronic lock. Everybody who wants to enter the room has to know the right key-code, which can change according to some pattern.

ACTOR STORY GENERAL STRUCTURE: The actor story which has to be constructed *as a symbolic structure* assumes that one can interpret the *real application case* as a *static* situation in the beginning which can *change* by some defined actions or events. Therefore an actor story will be organized as a sequence of static states connected by action or events causing changes.

TEXTUAL ACTOR STORY (TAS):

1. *State S1:* A worker is in a corridor.
2. *Action:* Walking along the corridor.
3. *State S2:* The worker has reached a door with a keypad. The door is closed.
4. *Action:* Move hand to keypad.
5. *State S3:* Hand is before the keypad.

6. *Action:* Enter a key-code.

7. *State S4:* The door is open. Behind the door is a room.

8. *Action:* Walking into the room.

9. *State S5:* The worker is in the work room.

In the real world there exists usually more than one action possibility. To cope with all possible cases one has to include these in the actor story as different continuations. By practical reasons it makes sense to limit the descriptions to those cases which are part of the solution announced in the vision statement.

4.2 Pictorial Mode Example

In this section a short example for a simple actor story in pictorial mode will be given. The *start state* is the same as in the example with the textual mode above.

PICTORIAL ACTOR STORY (PAS): A *pictorial actor story* is created with the following sequence of pictures:

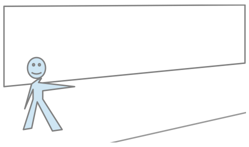

Figure 4.1: Worker in a corridor

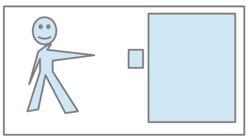

Figure 4.2: Worker before closed door with keypad

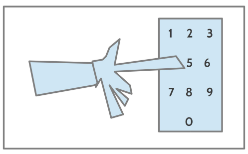

Figure 4.3: Hand at keypad

The drawings as such communicate some meaning, whose exact content depends from the viewer and his/ her learning history. To improve the communication of the meaning one can *unify* the pictorial and the textual mode into a *pictorial story* (not a comic!).

PICTORIAL-TEXTUAL LEXICON: It is an open question whether one should first construct a *pictorial-textual lexicon* and then generate a *unified pictorial-textual actor story* or vice versa. Because in this text it is assumed that the experts start with a pictorial and a textual actor story independently from each other it seems to be more naturally to take these two stories as starting point, align them in one unified multi-mode story and then derive from this

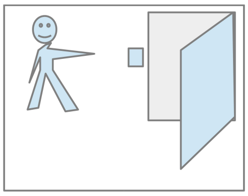

Figure 4.4: Door is open

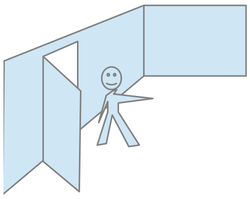

Figure 4.5: Worker is in the working room

unified story a possible lexicon. Proceeding in this way can reveal different points in both stories which seem not to be fully synchronized yet. This can help to refine the stories.

4.3 Unified Pictorial Textual Story (PTAS)

In this section a unified actor story is presented: unifying the textual and the pictorial mode without destroying the different parts.

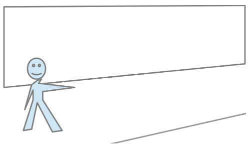

Figure 4.6: *State S1:* A worker is in a corridor. *Action:* Walking along the corridor.

4.4 Change in the Pictorial and Textual Mode

With the assumptions about the pictorial and the textual dimension of the world view at least one question expects more answers as given so far. In chapter 3 about the meaning of language an outline of the assumed cognitive structure of an expert has been presented. Based on the object structures in each expert which is included in the meaning function of an expression based language there is some minimal relationship between the pictorial and the textual mode. But zooming deeper into a pictorial actor story one is exposed not only to 'drawings as such'. Figure 4.11 indicates that the object structure of the cognitive structure *includes implicitly* a framework *like a 3-dimensional space*, within which all our objects are organized.[1,2] ,[3]

If one takes the final drawings in a pictorial actor story these show a *fixed perspective* into a *presupposed space*. The presupposed space allows many different perspectives but the drawings show only one of these many perspectives. The viewer of such a pictorial actor story is mimicking this pre-selected view. Extending this viewer perspective into the presupposed space one can infer an *embedded observer* in the *presupposed space*. This embedded observer has a certain position in this presupposed space, a certain direction and angle of viewing in this space. Thus if in this presup-

[2] Immanuel Kant. *Kritik der reinen Vernunft*. Johann Friedrich Hartknoch, Riga, 1 edition, 1781. There exist different modern critical editions in German as well as many translations in many languages

[3] Konrad Lorenz. Kants Lehre vom Apriorischen im Lichte gegenwärtiger Biologie. *Blätter für Deutsche Philosophie*, 15:94–125, 1941

[1] This reminds at the important work of Immanuel Kant (1781, 1787), in which he analyzed the conditions of human thinking. One of his insights was that human thinking is presupposing a spatial and timely structure in all its perceptions of reality. He called these conditions *transcendental conditions* because they are a kind of a pre-condition of all knowledge. Later modern science has proved these insights, but has shown even far more radical assumptions. A first connection between Kant's philosophy and modern evolutionary biology was worked out by the later Nobel prize winner Konrad Lorenz (1941).

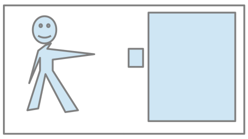

Figure 4.7: *State S2:* The worker has reached a door with a keypad. The door is closed. *Action:* Move hand to keypad.

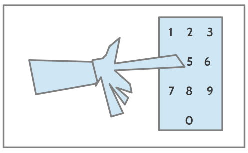

Figure 4.8: *State S3:* Hand is before the keypad.*Action:* Enter a key-code.

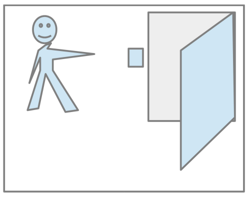

Figure 4.9: *State S4:* The door is open. Behind the door is a room. *Action:* Walking into the room.

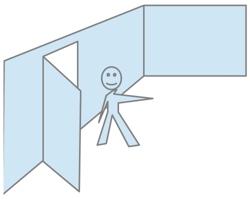

Figure 4.10: *State S5:* The worker is in the work room.

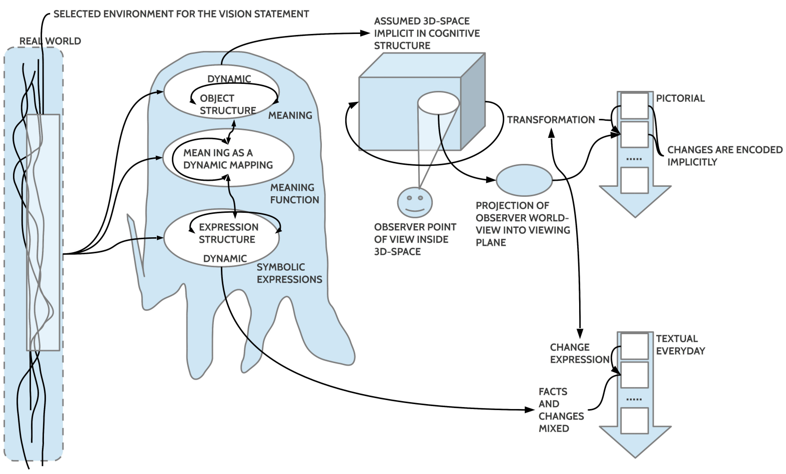

Figure 4.11: The cognitive dimensions begind the pictorial and textual modes

posed space things are *changing* then geometrically some *transformations* are occurring which can be perceived from the embedded observer. These transformations have a starting point and an ending point and it depends from the *timely resolution* how many *time slices* are representing the transformation. The individual drawings of the pictorial actor story represent either only the starting point and the endpoint of a transformation[4] or a more detailed *sequence of moments* from this transformation. Because one can assume that all human observers in the same 3-dimensional space have sufficient similar perceptions which will be stored in their object structure and which can be part of their meaning function one can assume that the textual mode which represents such perceived changes is rooted in this acquired knowledge and that it is this acquired knowledge which associates the changes encoded in sequences of drawings and encoded in symbolic expressions.

[4] which will be *perceived* from the observer as a *change*

5
Actor Story Modes: Mathematical

CONTEXT: In the preceding chapter a distinction between the *pictorial* and the *textual* mode of an actor story has been introduced, whereby the textual mode has been distinguished further by a textual mode with *everyday language* and a *mathematical language*. The last case that of a textual mode with a *mathematical* language will be illustrated in this chapter. In this chapter a further distinction will be made between the basic actor story (without actor models) and an extended actor story with additional actor models. While the actor models will be described in another chapter the interface between a basic actor story and possible actor models will be described in this text too.

The term *mathematical language* will in this text not yet be defined in an explicit way. Only an example using the mathematical language will be given. An explicit formal definition follows later.

5.1 Mathematical Meaning Function

A *mathematical actor story (MAS)* can be constructed like an *everyday textual actor story (TAS)* from scratch, but experience tells us, that it is helpful for a team, to have a TAS as starting point and as a point of reference for all.

Similar to the meaning function μ of the everyday language the mathematical language needs its own *meaning function* τ mapping the *object structure (OS)* into the mathematical expression structures and vice versa: $\tau : L_{expr} \leftrightarrow L_{mean}$. In this case $L_{expr} = L_{math}$ and $L_{mean} = OS$. If You compare the structure of the everyday meaning function μ with the mathematical meaning function τ then you can see that the 'meaning', the *object structure (OS)*, is in both functions the same. Both functions differ only in the kind of expressions which are associated with the object structures.

As explained before the meaning function τ is located in the assumed *cognitive machinery* of the actor.[1] Although the meaning function τ is *generally independent* from the meaning function μ, the existence of the meaning function μ on account of the *learning history* of the actor does influence the mathematical transformation τ in some way. The history of logic can shed some light on this hidden influence (cf. Kneale and Kneale (1962)[2]). Lacking such a complete story in this text some *informed guesses* will be made and then used.

[1] Often called *mind* somewhere in the brain, partially reconstructed by *psychology* based on the observable behavior and additionally motivated by phenomenological and physiological methods. It has to be stated that the whole story of these internal cognitive processes has to be written yet.

[2] William Knealy and Martha Kneale. *The Development of Logic*. Clarendon Press, Oxford (UK), 1 edition, 1962. Reprinted 1986 with corrections

EMPIRICAL AND COGNITIVE OBJECT STRUCTURE; SYMBOLIC REPRESENTATION: In the everyday textual version of the example actor story it is assumed that there exists an empirical situation eS1 which will be symbolically be represented as a *state S1* where a *worker* is in a *corridor*. The connecting 'link' between the empirical situation eS1 and the symbolic representation in the state S1 is an assumed learned cognitive object structure in the expert which allows the encoding of the empirical situation eS1 into the symbolic expressions of S1. The 'corridor' represents some special *spatial structure* and situated in this spatial structure is an *object* called 'worker'. While the expert with a presupposed meaning function μ for the everyday language encoding can translate the empirical situation eS1 in a textual actor story one has to assume that for the translation into a mathematical actor story there exists the same object structure as presupposed with μ but there is another meaning function τ which encodes these common object structure into other kinds of expressions, here into the expressions of a mathematical language L_{math}. These object structures can generally *vary* between different individuals on account of different learning histories, but they share some *common ground* on account of their relation to the same *causing world* outside of their brains. Assuming such a common object structure one can start with the assumption that the common object structure consists of *objects* which can be *named* like *W1* or *C1* and which can be subsumed to some *properties* like *WORKER(W1)* or *CORRIDOR(C1)*, where 'WORKER' and 'CORRIDOR' are also kinds of names for internally known sets of somethings which corresponds internally to these labels. One can try to *objectify* these internal somethings with the aid of *language games* as invented by the late Ludwig Wittgenstein during the years 1936 - 1946 (cf. Wittgenstein (1953)[3]), but this 'objectifying' does not substitute the internal structures 'corresponding' to the 'objective matters' between two different language users. In contrary, this 'objectifying' by playing language games induces those internal structures which serve then as the object structures for the individual meaning function τ, which have to be made as congruent as possible between different language users. Because there exists at least a *spatial relationship IS-IN()* between the two objects W1 and C1 one can name this relation too as *IS-IN(W1,C1)*. Again the 'meaning' of the expression 'IS-IN(W1,C1)' is only given by presupposing an internal object structure associated by a meaning function τ, where the internal structures are assumed to be correlated with some properties of the 'outer world'.

[3] Ludwig Wittgenstein. *Philosophical Investigations*. Macmillan, New York, 1953

KINDS OF EXPRESSIONS: So far there are three kinds of expressions indicating three kinds of meanings:

- *Names* of *objects*.
- *Name* of a *property* attached to *only one object*, called 1-ary predicate.
- *Name* of a *property* attached to *more than one object*, called n-ary predicate (n>1) or *n-ary relation*.

LANGUAGE GAMES AND MEANING: The case of *empirical measurement* is seen here as a special case of a language game. In an empirical measurement one compares some empirical phenomenon with an agreed *standard object* within an *defined procedure* which should – in the ideal case – yield a measurement result which is *independent* of the person which is doing the measurement and which is the same when it will be *repeated*. But one has to keep in mind that even in the case of an empirical measurement using a standard object the associated meaning is not given by those objective objects, properties or relations but by those processes, which internally are processing the perceived stimuli and set up in the individual actor his individual object structure which serves as the substrate of meaning. Thus even in the case that different actors are using the same measurement procedure it can happen that these different actors are setting up different internal object structures and thereby they are setting up different internal meanings. This difference results from the interaction of perception and already given experience/ knowledge, which is used to embed perceptual structures into other structures. [4,5]

[4] If you will look for books dealing with measurement you will encounter some difficulty: books about measurement usually do not discuss the philosophical conditions which have to be fulfilled to enable a measurement. As an example see Krantz et.al. (1971)

[5] David H. Krantz, R.Duncan Luce, Patrick Suppes, and Amos Tversky. Foundations of Measurement. Volume I. Additive and Polynomial Representations, 1971

5.2 State Description

As illustrated before we have three kinds of expressions with associated three different kinds of meanings, which can differ between different experts. Every expression combining predicates or relations with names of objects is here called a *statement of fact* or in brief a *fact (F)*. And a *state (S)* is understood here as nothing else then a *set of facts*. Thus the state S1 of the example can be written as: $S1 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1)\}$. A *state* as a collection of facts is therefore a *set of expressions* of a certain kind. As assumed before these expressions presuppose a *meaning function* τ which associates each expression with some meaning, which in the case of an actor story is supposed to *correspond* to some *empirical situation (eS)* where one can identify those kinds of objects, properties, and relations which are represented by the expressions of a certain state. If an actor can associate a certain fact f of a state S with a meaning m by his meaning function τ and the actor can associate this meaning m with some perception of an empirical situation eS then one can classify the fact f to be *true* with regard to the meaning function τ, to the meaning and to the perceived empirical properties. Otherwise the fact is *not true* or *undecidable*. For an actor story it is assumed that it describes a sequence of states whose facts are all true with regard to a supposed situation.

5.3 Objects and Actors

Already the before described simple state description points to *different kinds of objects*: those objects which are in some sense *passive* objects, which do not act by themselves, and those objects which can be *active*, which can respond to events in their environment. Those active objects are called in this text *actors*. And there is even a further distinction: an active actor can occur as an *individual, concrete object* like a human person, an animal, a robot, but there are also actors which induce effects in a situation which are *not* individual, concrete objects but are *non-individual, abstract* objects like 'temperature', 'humidity of the air', 'noise of a city', 'weather' or the 'climate' or something like this. Like in every empirical theory these abstract objects are represented by a *name* which functions like a *theoretical term* whose concrete meaning is given by some *concrete effects* which can be associated with such a term.[6,7] Thus as long as there exists at least one observable effect one can define at least an abstract actor. If there exists additionally a concrete object-like structure one can define an individual, concrete actor.

[6] For a broad overview of the meaning of *theoretical terms* in empirical theories see Frederick Suppe (1977), p.27ff

[7] F. Suppe, editor. *The Structure of Scientific Theories*. University of Illinois Press, Urbana, 2 edition, 1979

5.4 Change Statement

5.4.1 Basic Change Statement

As discussed before in the example with the pictorial actor story one can describe a *change* only indirectly by the *differences* between two consecutive states. In the *everyday textual version* one can use an expression to talk directly about a change but the meaning of this change expression is rooted back trough the meaning function μ to those object structures which represent changes. These *change-relevant structures* contain finite sequences of consecutive situations which can be addressed by a language expression without the need that the language expressions show the properties of the meaning in the format of the expression.

In the example with a *mathematical textual actor story* there exists a mixture of both aspects: as in the case of the pictorial actor story one has (i) two *states* S1 and S2 with different sets of facts, and (ii) a *change statement* which talks about the change. To illustrate this I construct state S2 of the example first. Afterwards I show how one can generate S2 out of S1.

The facts of state S1 are assumed to be still valid, but some more facts have to be added. A *mathematical change expression* can manipulate these facts either by *deleting* some given facts or by *creating* some new facts. In the actual example there is an *individual, concrete actor* named 'W1' which can cause an *effect* by the action *walking()*. Here 'walking()' is the name of a kind of change where the actor 'W1' applies an action 'onto himself' by changing his position in space. The effect of this walking-action is described by the following new facts:

$X+ = \{EDOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), PART - OF(K1, D1), BEFORE(W1, D1)\}$.

The complete new state S2 is then given by the formula

$S2 = S1 - (X-) \cup (X+)$.

Because 'X-' is empty we have only 'X+' and therefore we have the old state S1 *unified* with the set 'X+'. Thus the new empirical situation eS2 will be described by the state S2 with the facts

$\{WORKER(W1), CORRIDOR(C1), IS - IN(W1, C1), EDOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), PART - OF(K1, D1), BEFORE(W1, D1)\}$.

Given the *beginning* of the change (= S1) and the *end* of the change (=S2) one can see how one can construct an appropriate change rule. In this text the following pattern for a change-rule is adopted:

$\langle S1, S2, W1, walking(), \{X-, X+\}\rangle$

This is a *meta-rule* talking *about* the states S1 and S2, about the actor

'W1', about the *action* 'walking()' which is realized by the actor, and about the *observable* facts, those which will be *deleted* in S1 and those which will be *created* as new facts in S2.

In the normal mode of constructing an actor story state S1 is given and state S2 has to be constructed with the aid of a change rule. In the above example the assumed action is named 'walk()'. Because the effect of the action 'walk()' is completely specified by the sets 'X-' and 'X+' of the change-rule pattern a further specification of the action 'walk()' is not necessary. With the specifications:

$X- = \{\}$ (The empty set $\emptyset$)

$X+ = \{EDOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), PART-OF(K1, D1), BEFORE(W1, D1)\}$.

one can construct S2 according to the schema $S2 = S1 - (X-) \cup (X+)$'completely.

5.4.2 Advanced Change Statement

In the above section the action of the change statement is not further specified because the effect of the action is completely given. In case of an actor story description this is acceptable because the actor story describes only observable facts (thereby assuming a 3rd person point of view).

CHANGE BY A BEHAVIOR FUNCTION: As mentioned in the introductory chapter 1 one can extend an ordinary actor story by so-called *actor models (AMs)* which have additional specifications of a *behavior function* ϕ which computes to an *input* value x some *output* value y like $\phi(x) = y$. In that case the action has to be specified in accordance with the definition of the behavior function. This changes the pattern of the change statement as follows:

$\langle S1, S2, W1, walking(x), y\rangle$

In this pattern the action 'walking(x)' is an action of the actor 'W1' and 'W1' is assumed to have an actor model $\langle W1, x, y, walk\rangle$ with $walk : x \longmapsto y$. The name 'x' represents the different kinds of *input* values which are possible for the *behavior function* $\phi = walk$ and 'y' represents the sets 'X-' and 'X+' describing the *output* of the behavior function. The set of all output values of a fully specified behavior function can contain more values then those which are specified in the base version of an actor story. Because it is required that the output values of a behavior function are in *full agreement* with the given actor story there exists the minimal condition that all values $V = (X-) \cup (X+)$ specified in the given actor story have to be contained in the range of the specified behavior function. This means

$V \subseteq ran(\phi)$

CHANGE BY AN EMBEDDED BEHAVIOR FUNCTION: If one has already defined an actor model with a behavior function ϕ then it can happen that an actor realizes an action with an object, which itself is an actor. This can happen if there exists an assisting actor with an interface and with a behavior function. In this simple example we have the case of an electronic door (see below) with a keypad as interface and a specified behavior function. Thus the behavior function of the electronic door expects certain numbers as input to open the closed door. Therefore if the executing actor of the example story touches the keypad and enters a sequence of numbers to satisfy the expected pattern of numbers then this sequence of numbers is the input for the internal behavior function of the electronic door which computes from this input its output: depending from input open the door or stay closed.

5.4.3 The Whole Story

With these considerations one can construct the actor story in mathematical mode as follows:

1. *State S1:* A worker is in a corridor. $S1 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1)\}$
2. *Action:* Walking along the corridor. $\langle S1, S2, W1, walk(), X- = \{\}, X+ = \{DOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1), HAND(H1), PART-OF(H1, W1), AT-BODY(H1, W1)\}$.
3. *State S2:* The worker has reached a door with a keypad. The door is closed. $S2 = S1 - (X-) \cup (X+)$, $S2 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1) DOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1), HAND(H1), PART-OF(H1, W1), AT-BODY(H1, W1)\}$
4. *Action:* Move hand to keypad. $\langle S2, S3, W1, moveHand(), X- = \{AT-BODY(H1)\}, X+ = \{BEFORE(H1, K1)\}\rangle$
5. *State S3:* Hand is before the keypad. $S3 = S2 - (X-) \cup (X+) = \{WORKER(W1), HAND(H1), PART-OF(H1, W1), BEFORE(H1, K1), CORRIDOR(C1), IS-IN(W1, C1) DOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1)\}$
6. *Action:* Enter a key-code. $\langle S3, S4, W1, enterCode(K1, \langle 7,7,5,7\rangle), X- = \{CLOSED(D1), BEFORE(H1, K1)\}$, $X+ = \{AT-BODY(H1, W1), OPEN(D1), ROOM(R1), BEHIND(R1, D1)\}\rangle$
   /* This is the case of an embedded behavior function, which has to be served by the action of the executing actor. */
7. *State S4:* The door is open. Behind the door is a room. $S4 = S3 - (X-) \cup (X+) = \{WORKER(W1), HAND(H1), PART-OF(H1, W1), AT-BODY(H1, W1), CORRIDOR(C1), IS-IN(W1, C1), DOOR(D1), OPEN(D1), KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1), WORK-ROOM(R1), BEHIND(R1, D1)\}$
8. *Action:* Walking into the room. $\langle S4, S5, W1, walk(), X- = \{CORRIDOR(C1), IS-IN(W1, C1), BEFORE(W1, D1), KEYPAD(K1), BESIDES(K1, D1), BEHIND(R1, D1)\}, X+ = \{IS-IN(W1, R1)\}\rangle$
9. *State S5:* The worker is in the work room. $S5 = S4 - (X-) \cup (X+) = \{WORKER(W1), HAND(H1), PART-OF(H1, W1), AT-BODY(H1, W1), DOOR(D1), OPEN(D1), WORK-ROOM(R1), IS-IN(W1, R1)\}$

5.5 Basic or Advanced Change Encoding

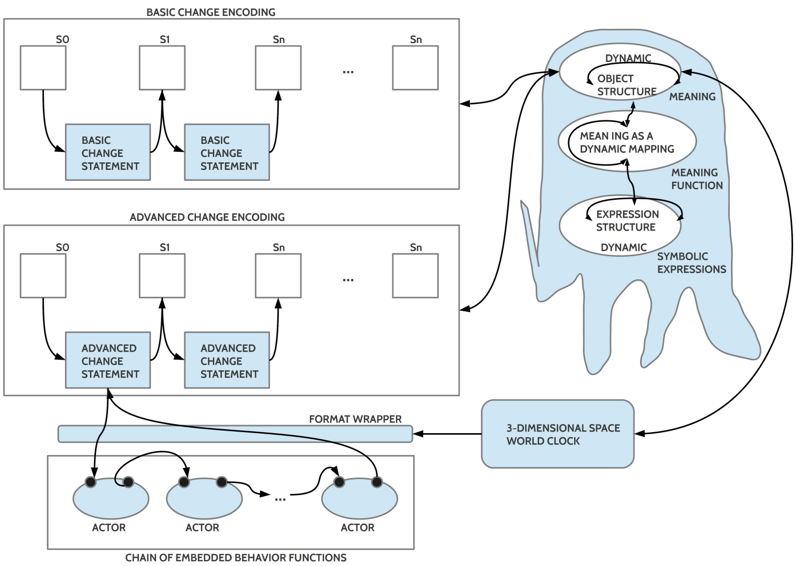

Figure 5.1: Difference of basic and advanced change encoding

After the introduction of the different change encodings figure 5.1 illustrates the hidden structures which are 'at work' in these encodings.

The most simple case is a *mathematical actor story (MAS)* which uses only *basic change statements*. In this case the action which causes the change is named but has no further specifications. Only the desired changes are explicitly given. This case illustrates a basic MAS which usually will be used in the beginning of the analysis, following the preceding pictorial and textual actor story versions.

As soon as one wants to analyze the actor story with more details, especially with regard to the participating actors and their behavior functions, it will be necessary to extend the actor story with *actor models (AMs)* with their *behavior functions* ϕ which have all the format of mappings ϕ : $IN \longmapsto OUT$. If one wants to address these behavior functions explicitly then one has to specify the action encoding in the change statement in a way that the intended behavior function of a certain actor can be identified. Thus, if we have the *basic* change statement given in the format:

$\langle S, S', actor, action(), default\ change\ effect\rangle$

then we have to extend this format into an *advanced* change statement as follows:

$\langle S, S', actor, action(embedded\ actor\ name, IN), OUT = default\ change\ effect\rangle$

The *calling actor* will apply an action, but in this case this action is addressing an *embedded actor* and delivers all the parameters which are necessary for the input (IN) of the embedded actor. The change effect will then be computed from the embedded actor by his behavior function and this dynamically computed output (OUT) will *overwrite* the default change effect. The case of the embedded actor includes the possibility to extend the computation by a chain of embedded actors where the first one calls another one and only the last embedded actor in this chain will then respond with his output to the calling change statement.

It has to keep in mind that there can be *more than one change statement* associated with one actual state.

Another important point which should be mentioned here without going into the details of it is the fact that a mathematical actor story presupposes some *space model* (usually a 3-dimensional one if dealing with cognitive matters) and a *time model*. Thus there has to be assumed an *implicit encoding* of all inputs and outputs with regard to these implicitly assumed structures.

6
Actor Model

CONTEXT: In the preceding chapter 5 it has been explained that an advanced change statement can include the call to an object which is an actor having a behavior function which enables the actor to respond to an input with an output in a specific way determined by the behavior function. To use such an actor with a behavior function one has to specify such an actor. Such a specification is here called an *actor model (AM)* or an *actor (A)*.

6.1 Basic Definitions of Systems

INPUT-OUTPUT SYSTEMS: In the general case an actor object is an *input-output system (IOS)* which is interacting with its environment by the *inputs (I)* and *outputs (O)*, controlled by a *behavior function* ϕ. The general format of an input-output system is given by the definition:

$$IOS(x) \; iff \; x = \langle I, O, \phi \rangle \tag{6.1}$$

$$\phi : I \longmapsto O \tag{6.2}$$

The *input* and *output* of an input-output system has *to be in accordance* with the *facts (F)* of an actor story. While the specification of an actor story is only finite and the amount of outputs of a behavior function can be infinite it is the duty of the experts and the stakeholder to classify the *surplus* of the generated output as still in accordance with the *intention* of the actor story or not, stating that $F \subseteq rn(\phi)$.

LEARNING (ADAPTIVE) IO-SYSTEM: The general definition of input-output Systems does not explicitly specify whether a system is adaptive or learning. If one wants to use some learning capacity explicitly then one has to give the learning function a format which allows to talk about *learning* explicitly. This text uses the following format:

$$LIOS(x) \; iff \; x = \langle I, O, IS, \phi \rangle \tag{6.3}$$

$$\phi : I \times IS \longmapsto IS \times O \tag{6.4}$$

Thus if the internal states IS of an actor would include a *memory* $M \subseteq IS$ with some content – e.g. '0' –, then the behavior function ϕ would react

– if using the memory at all – with the actually given content – e.g. '0' –. But in the response the behavior function could change the content of the memory to some different value – e.g. '1' – and then, the next time when the behavior function has to respond to some input by using the memory it could find some new content – e.g. a '1' instead of a '0'–.

ORGANIZED COMPLEXITY: Learning IO-systems can change their behavior depending from their input. If one has *collections* of learning systems then these collections constitute a new system whose elements are systems where each system can become dependent from each other system. Warren Weaver (1958) calls such collective systems an *organized complexity* on account of this inter dependencies of all member systems.[1] Collections of systems where the member systems are not dependent from each other are called a *disorganized complexity*.

[1] Warren Weaver. A quarter century in the natural sciences. In *Rockefeller Foundation Annual Report, 1958*, pages 7 – 15. Rockefeller Foundation, 1958. URL: https://assets.rockefellerfoundation.org/app/uploads/2015053012 Report-1958.pdf

STRONG EMERGENCE: Organized complex systems show an effect which is called *emergence*. The intuitive meaning is that the behavior ϕ_i of the individual systems can be different from the behavior Φ of the whole set of systems. But despite of the phenomenal *difference* there exists an inherent *dependency* of the general behavior Φ from the individual behavior ϕ_i of the individual systems. *Reducing* the collection of systems into a mere set of disconnected individual systems (a disorganized complexity) would destroy the overall behavior Φ. Disorganized complexities can nevertheless show some macro behavior which is different from the behavior of the individual elements but these macro phenomena are *not dependent* from the behavior of the individual systems. To distinguish both cases this texts speaks of a *weak* emergence in case of macro effects with a disorganized complexity and of a *strong* emergence in case of a organized complexity.

COMPLEX SYSTEM SCIENCES: The phenomenon of strong emergence constitutes a kind of research subject which can not be reduced to more simpler cases without destroying the subject itself. Famous examples of strong emergence can be found in the realm of biological systems: every *body* is a galaxy of individual cells which are cooperating in intriguing ways to enable complex functions to keep a whole body functioning. But *groups* of plants, animals and especially of exemplars of the homo sapiens life form constitute *new kinds of collections* whose interactions constitute qualitatively new behavior beyond that what an individual member of this group could do. Nevertheless this overall behavior is completely dependent from the individual members. To investigate such overall behavior is a science on its own (biology, sociology, cultural anthropology, ...).

COMPLEX SYSTEMS TOOL: From these considerations it follows that the combination of actor story and actor model allows the construction of arbitrary complex systems with strong emergence.

ACTOR AND INTERFACE: Usually one is talking about actors as if these would occur in a situation as a whole. But if you look closer then it becomes evident that what you primarily can observe is only the *surface* of an actor which can receive *inputs* and can respond with *outputs*. Possible internal states (IS) or the working behavior function ϕ is not observable. You can only try to infer these from the outside by observing the different inputs (I) and outputs (O) to get an *empirical behavior function* ϕ_e like $phi_e = \{(I_1, O_1), I_2, O_2), ..., I_n, O_n)\}$ with $\phi_e \subseteq \phi$ where ϕ represents a possible formally defined behavior function.

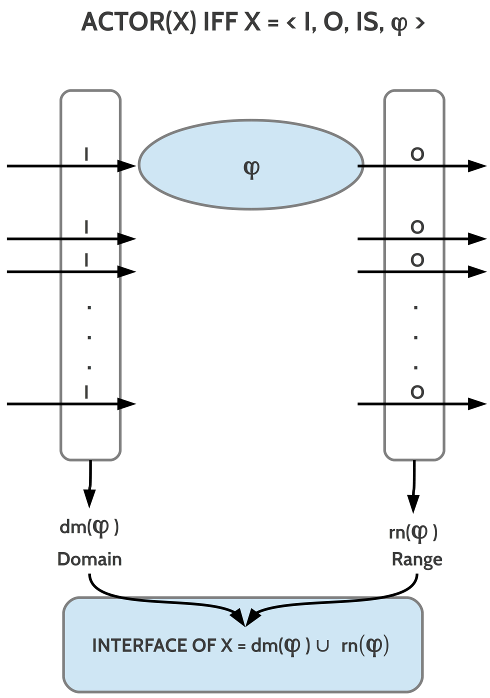

Figure 6.1: The interface of an actor is realized by the unified set of possible inputs and outputs

In a real situation S_e corresponding to a state S in an actor story one has to differentiate even further between those *qualities of the environment* of an actor A which can function as an input event and those *parts of the interface* which can *receive inputs* from the environment. If one classifies inputs as *continuous* qualities like 'temperature', 'air pressure', 'noise' or *short-time* qualities which occur as *events* like a 'push', an 'acoustic signal', a 'change in color' etc. then one has further to define those *parts of an interface* which are specific to *sense* these inputs. Thus one has to distinguish the kind of *input stimuli (Σ)* which can hit an interface and the kind of *interface input sensors (SENS)* which can *sense* the input. Therefore the specification of the input part of an interface should typically include a table of kinds of input events associated with kinds of input sensors able to sense these input events. The same has to be assumed for the output part of an interface. Here we have to assume different kinds of *output actuators (OUT)* which can cause certain *kinds of responses* ϱ which represent effects in the environment. The same kind of actuator can cause different kinds of effects depending from the environment (e.g. being on the surface of the earth, being under water, being in free space,...).

From this one can infer that the description of the observable interface of an actor is necessarily *incomplete* compared to a purely mathematical description. But for the later *realization* of an *empirical* version of the interface one needs such a concrete specification.

ACTORS: HIDDEN AND OBSERVABLE: Following this line of thought by distinguishing the *surface* of an actor with its *input* and *output elements* it follows that the *interface* of an actor becomes the observable, visible part of a *local actor* which is directly involved in the *interactions* with the *environment* of a state.

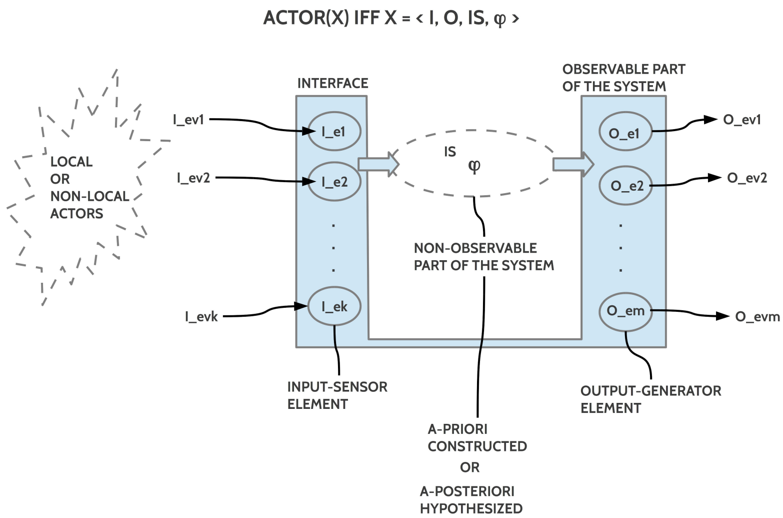

Figure 6.2: Hidden actor inside, observable actor outside (surface, interface)

This distinction of *input elements* I_e sensing certain kinds of triggering events induces the necessity to identify the *different kinds of causing input events* I_{ev}. Without an explicit knowledge of such input events it would not be possible to describe a state sufficiently well.

For the description of an actor in the context of interactions it follows, that one should distinguish between the *observable interface* with an observable empirical behavior function ϕ_e and the 'rest' which contains possible internal states IS and the theoretical behavior function ϕ, which *is not observable*! The behavior function in the system description is always a *hypothesis*, a *theoretical construct*, which is based on the *empirical behavior function* with $\phi_e \subseteq \phi$.

$$AM(x)\ iff\ x = \langle INF, IS, \phi \rangle \tag{6.5}$$

$$INF := \langle I, O \rangle \tag{6.6}$$

$$\phi : I \times IS \longmapsto IS \times O \tag{6.7}$$

6.2 Actor Story Refinements

OUTLINE OR DETAILED AS: With the ability to describe an actor with an interface $INTF$ addressing explicit sensor and actuator elements one has the *choice* between two kinds of an actor story description: an actor story as an *outline* or an actor story with an *explicit specification* of all interface elements. This will be demonstrated in some examples below.

EXAMPLE SKETCHY DESCRIPTION: A state S1 is outlined as follows: "A worker is in a corridor". Translated into a formalized version: $S1 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1)\}$. A rough description of an action combined with the effects can be given as follows (textual: "The worker is walking along the corridor.") in a formal mode: $\langle S1, S2, W1, walk(), X- = \{\}, X+ = \{EDOOR(D1), PART-OF(D1, C1), CLOSED(D1), KEYPAD(K1),$
$BESIDES(K1, D1), POSITION(W1, C1, P2), BEFORE(W1, D1)\}$.

Here an *activity* is reported which is generated by the actor W1 and which is acting on the object C1. Assuming tacitly that the reader knows the hidden meaning of this statement then it is sufficient to describe only the intended effects (appearing before an electronic door which his closed).

EXAMPLE DETAILED ACTION DESCRIPTION: If one wants to specify in more Details the action *walk()* then one has to specify the interface elements of the worker W1 which can cause an action called 'walking' and one has to specify the interface of the environment with those elements, which can sense the action. In this simple example the interface element of the worker W1 is given by the actuator elements 'legs' and the effect of a walk action using the legs is with regard to the corridor object C1 with its interface input elements 'position'. Walking changes the position in the corridor. A possible formalization could look like this:

$S1 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1), POSITION(W1, C1, P1),$
$INTF(I1), PART-OF(I1, W1), LEGS(L1), PART-OF(L1, I1)\}$
$\langle S1, S2, W1, walk(L1, C1, P1), X- = \{POSITION(W1, C1, P1)\}, X+ =$
$\{POSITION(W1, C1, P2), EDOOR(D1), PART-OF(D1, C1), CLOSED(D1),$
$KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1)\}$.

The resulting follow-up state S2 is then given by the formula:

State S2 in textual mode: The worker has reached a door with a keypad. The door is closed.

In formal mode: $S2 = S1 - (X-) \cup (X+)$, which reads as:

$S2 = \{WORKER(W1), CORRIDOR(C1), IS-IN(W1, C1), POSITION(W1, C1, P2),$
$EDOOR(D1), CLOSED(D1), KEYPAD(K1), BESIDES(K1, D1), BEFORE(W1, D1)\}$

Bibliography

[BK08] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. MIT Press, Cambridge (MA), 1th edition, 2008.

[Blo00] Paul Bloom. *How Children Learn the Meanings of Words*. The MIT Press, Cambridge (MA), 2000.

[DHW07] Gerd Doeben-Henisch and Matthias Wagner. Validation within Safety Critical Systems Engineering from a Computational Semiotics Point of View. In *Proceedings of the IEEE Africon2007 Conference*. IEEE, 2007.

[FM95] Paul Fletcher and Brian MacWhinney, editors. *The Handbook of Child Language*. Basil Blackwell Ltd, Oxford (UK), 1995.

[Goe31] Kurt Goedel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, i. *Monatshefte fuer Mathematik und Physik*, 38:173–98, 1931.

[Gru08] Jonathan Grudin. A Moving Target: The Evolution of HCI. In A. Sears and J.A. Jacko, editors, *The Human-Computer Interaction Handbook. Fundamentals, Evolving Technologies, and emerging Applications*. 2 edition, 2008.

[Har92] Margaret Harris, editor. *Language Experience and Language Development: From Input to Uptake*. Lawrence Erlbaum Associates, Hillsdale (USA), 1992.

[Haw02] Stephen Hawking. Gödel and the end of physics, 2002. http://www.hawking.org.uk/godel-and-the-end-of-physics.html . Another URL with the audio file: http://www.damtp.cam.ac.uk/events/strings02/dirac/hawking/.

[Kan81] Immanuel Kant. *Kritik der reinen Vernunft*. Johann Friedrich Hartknoch, Riga, 1 edition, 1781. There exist different modern critical editions in German as well as many translations in many languages.

[KK62] William Knealy and Martha Kneale. *The Development of Logic*. Clarendon Press, Oxford (UK), 1 edition, 1962. Reprinted 1986 with corrections.

[KLST71] David H. Krantz, R.Duncan Luce, Patrick Suppes, and Amos Tversky. Foundations of Measurement. Volume I. Additive and Polynomial Representations, 1971.

[Kra19] Thomas Krause, Johannes; Trappe. *Die Reise unserer Gene: Eine Geschichte über uns und unsere Vorfahren*. Ullstein Buchverlag, Berlin, 5th edition, 2019.

[Lor41] Konrad Lorenz. Kants Lehre vom Apriorischen im Lichte gegenwärtiger Biologie. *Blätter für Deutsche Philosophie*, 15:94–125, 1941.

[MP07] John H. Miller and Scott E. Page. *Complex Adaptive Systems. Introduction to Computational Models of Social Life*. Princeton University Press, Princeton - Oxford, 1 edition, 2007.

[Pew03] Richard W. Pew. Introduction. Evolution of human-computer interaction: From memex to bluetooth and beyond. In J.A. Jacko and A. Sears, editors, *The Human-Computer Interaction Handbook. Fundamentals, Evolving Technologies, and emerging Applications*. 1 edition, 2003.

[Sup79] F. Suppe, editor. *The Structure of Scientific Theories*. University of Illinois Press, Urbana, 2 edition, 1979.

[Tur 7] Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42(2):230–265, 1936-7. http://www.turingarchive.org/browse.php/B/12.

[Wea58] Warren Weaver. A quarter century in the natural sciences. In *Rockefeller Foundation Annual Report, 1958*, pages 7 – 15. Rockefeller Foundation, 1958. URL: https://assets.rockefellerfoundation.org/app/uploads/20150530122220/Annual-Report-1958.pdf.

[Wit53] Ludwig Wittgenstein. *Philosophical Investigations*. Macmillan, New York, 1953.

Index