GERD DOEBEN-HENISCH

ACTOR ACTOR INTER-ACTION [AAI]

NOVEMBER 7-13, 2019 - VERSION 15.03

UFFMM.ORG

Copyright © 2019 Gerd Doeben-Henisch

PUBLISHED BY UFFMM.ORG

UFFMM.ORG, ISSN 2567-6458

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means except for brief quotations in printed reviews, without the prior permission of the publisher.

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 AAI Course Program: This text presents a short, condensed version of an analysis using the AAI (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 AAI 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/.

Terminology: HCI - HMI - AAI From the history of computer after the World War II[1] 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).*

[1] 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

1
The 'All in One View'

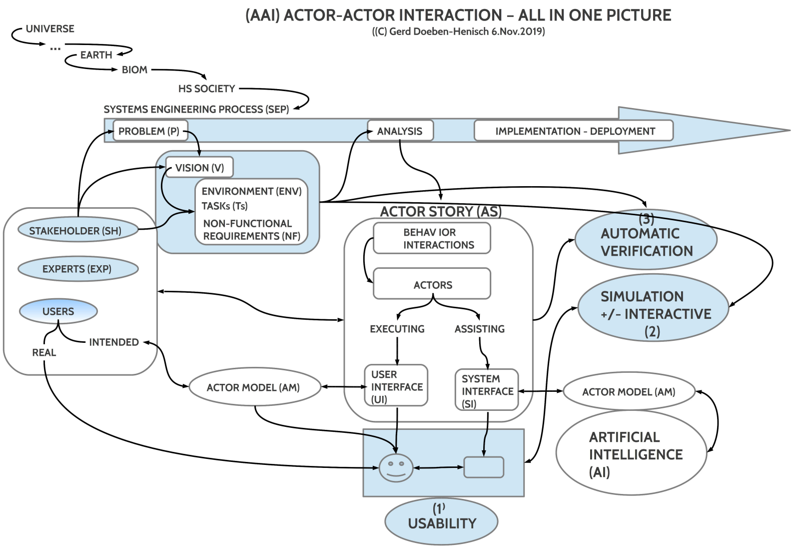

Figure 1.1: AAI analysis, the 'All in One View'

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

FIND A SOLUTION: The whole machinery of the *Actor-Actor Interaction Analysis* – short: AAI 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 represent 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 step wise into a more concrete description, such that the description worked out within an AAI 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.

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[4](cf. Krause et.al. (2019) [5]) 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 and 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 AAI 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.

[4] '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.

[5] 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.[6] 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 .

[6] 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[7]; 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.
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.

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

IMPLEMENTATION: The next phase in the systems engineering process after the AAI 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 from an usability test, from a simulation and from an automatic verification process.

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, but 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 logic 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. But the language expression as such, the sound which one can hear, is also not the pointer![1] The pointer from language expression to something else is located in our brain.

[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.

All children have to *learn* step by step which kinds of language expressions have to be *associated* with which kinds of other things 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. 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. Thus the meaning function represents something like 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)[2], Fletcher and MacWhinney (1995)[3], and Bloom (2000)[4]). Thus 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] Margaret Harris, editor. *Language Experience and Language Development: From Input to Uptake*. Lawrence Erlbaum Associates, Hillsdale (USA), 1992

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

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

Thus 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 *synaptic gap* which is a steady source of misunderstandings and errors caused by this gap (cf. Doeben-Henisch and Wagner (2007)[5]).

[5] 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

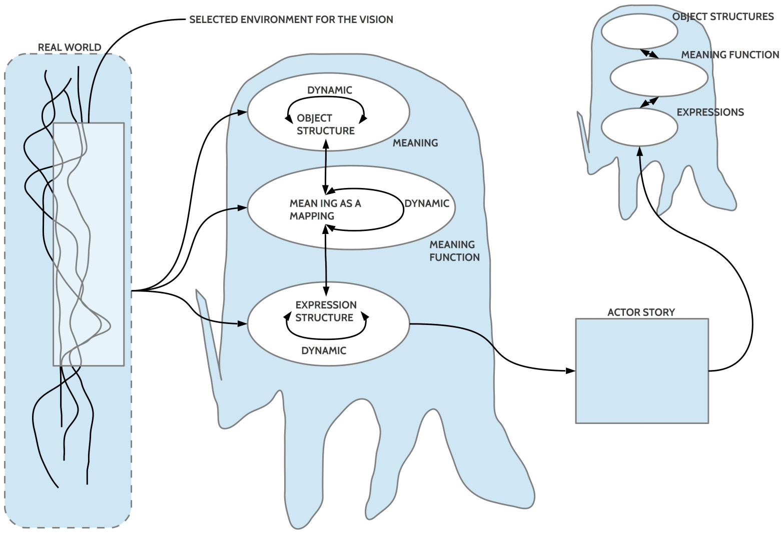

Figure 3.1: Minimal assumptions about the interacting language related systems in the head of the participants

ACTOR STORY AND MEANING: 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 using *everyday language* the different experts can always make small tests whether this assumption of the *sufficient same meaning* is *true* while 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.

If you would instead of the normal everyday language use the kind of language used in *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 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* style differentiated from the *textual* style. Finally if one wants to apply more advanced mathematical procedures and even computational processes using a computer it must be possible to write an actor story as a formal symbolic structure. But then the question arises 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.

The *primary assumption* is given with the basic cognitive systems for object structures, expression structures, 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 part 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 (LT-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. The *static* objects, properties and relations can be represented directly by drawings and by written expressions. *Changes* can be processed by drawings only implicitly

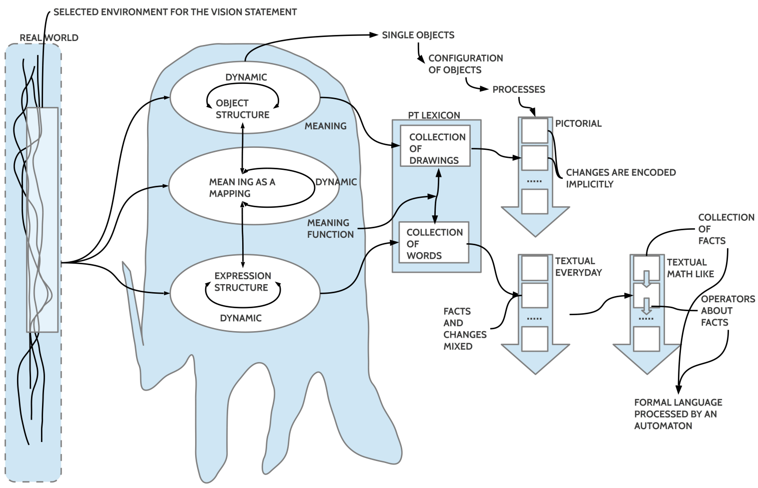

Figure 3.2: Different kinds of actor stories with regard to meaning

by successive pictures which are different. With written expressions one can use special expressions talking about actions and changes, but the meaning of these expressions is also not directly encoded but only indirectly by referring to static objects and properties in a situation and telling that they are changing.

Thus if you say that 'the door opens', 'he moves his arms', 'the apple falls down', ... and the like then you are presupposing that everybody can imagine by his own experience what it means that a 'door opens', an 'arm moves', an apple falls down'. This different nature of *static* and *dynamic* meanings becomes very clear if one constructs a formal representation of a process. In the mathematical representation used in this text the mathematical representation is also a collection of expressions but located on two different levels. On the *object level* one can assemble collections of *fact expressions* describing a *static* configuration of objects constituting a situation, a *state*. Like in the pictorial mode *changes* can only be expressed in the mathematical mode as differences between two consecutive states. 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.

Thus in the mathematical mode one can introduce formal expressions on the *change level*. 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 a textual encoding the meaning of changes is depending from the meaning of the fact expressions and what is known about their possible changes. Nevertheless, as soon as such formal encodings of object and change expressions are realized one can look to these formal expressions as being 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.

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.

[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.

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

[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.

Index