KOMEGA REQUIREMENTS No.4, Version 5
Basic Application Scenario
*

Gerd Doeben-Henisch (gerd@doeben-henisch)
in discussion with
Athene Sorokoswi [AS] (athene.sorokowski@gmx.de)
Philipp Westermeier [PW] (p.westermeier@mailbox.org)

September 13, 2020

Abstract

This text describes the basic requirements for the komega software project, which is part of a larger project in the domain of an applied cultural anthropology. This is version 5 of the basic requirements No.4 which replaces No.4-v4. More information to the actual theoretical background can be found in the posts with the title 'Extended Concept for Meaning Based Inferences – Part 2. Version 2'[1] as well as in the post with the title 'Actor Epistemology and Semiotics. Version 1'.[2]. The modifications in this version 5 compared to version 4 are due to the ongoing implementation of the theory in python[3] and are related to the first version of a simple simulator cycle.

Contents

[1]See: `https://www.uffmm.org/2020/09/02/extended-concept-for-meaning-based-inferences-part-2-version`

[2]https://www.uffmm.org/2020/09/03/actor-epistemology-and-semiotics-version-1/

[3]See `https://www.uffmm.org/2019/04/01/co-learning-with-python-3/`

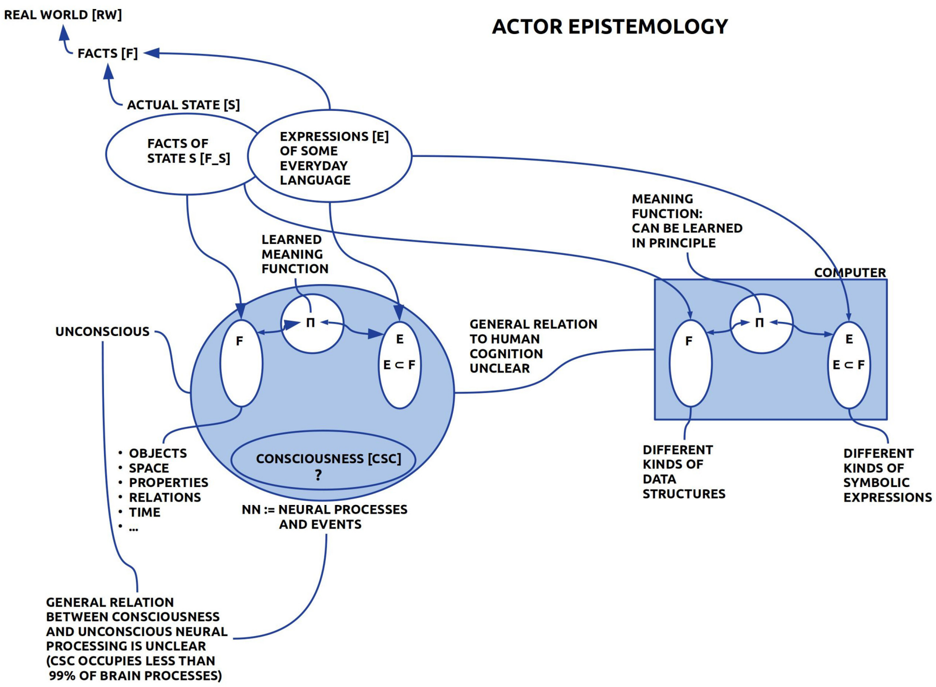

Figure 1: Actor epistemology as main point of view behind the project

1 Actor Epistemology and Semiotics

The task of this text is to describe the requirements which are framing the coding of the theory. Figure 1 delineates the theoretical point of view behind the whole project. The main actor is not the computer but the *human actor* HA. The reason behind this setting is the fact that the human actor is the only real bearer of knowledge on this planet. And this human knowledge has the specialty that it is organized in at least two main dimensions: the knowledge of the real structures of the world – including the own system – F which is completely encoded in neural states NN of the brain, and as part of this basic dimension there is a special subset of knowledge functioning as expressions E of a symbolic language, which are related by an *adaptive meaning function* μ to

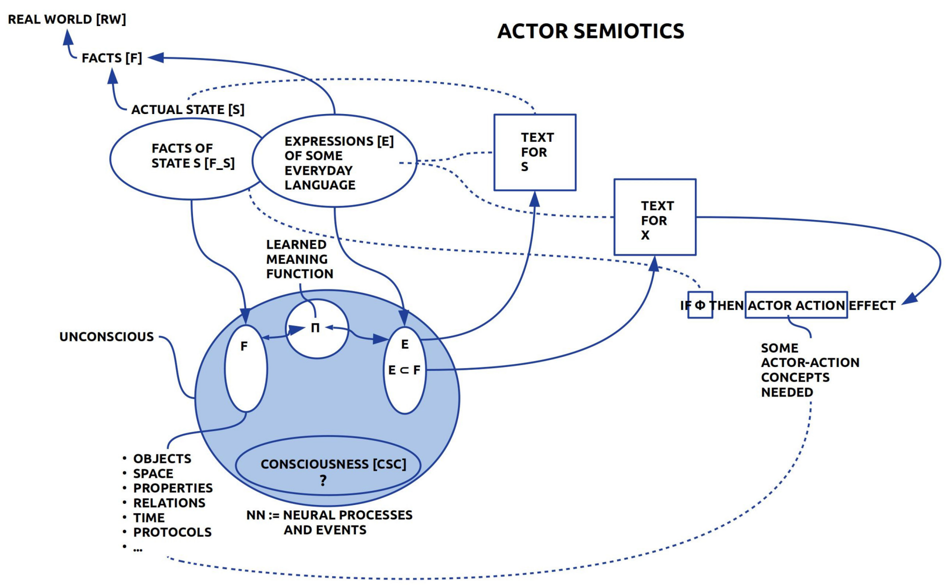

Figure 2: Actor semiotics points to the expressions used by human actors in their communications which are functioning as meaningful signs enabled by a working meaning function

parts of the knowledge. Communication between different brains is only possible by the expressions E which are transformed into real world events E_{RW} which can be exchanged between different bodies. Without the human actor with its built-in and *individually learned* meaning function μ these expressions are without meaning. Therefore any kind of meaningful communication about the real world has to put the human actor in the center of the discourse. Computer can be used to support such a meaningful communication but only in a very limited and specialized way. Nevertheless *as real part of real human communication* computer can be of central importance in the future.

If two different brains want to communicate with each other they have to use expressions E_{RW} embedded in a shared meaning function μ. These expressions can be organized as *texts* TXT which represent a finite set of expressions organized in a linear sequence.

In this project only two kinds of texts will be assumed: First a text which is describing a *state* S of the *real world* RW, either as *actually given* or as *assumed to be possible in the future*. It is assumed that the expressions of such a state description S describe so-called *facts* F_{RW} of the real world, which can be observed and which can be decided by the authors of the text as *being the case*. 'Being the case' is either inferred from these expressions *directly* by exploiting the meaning function μ or these expressions can be *transformed* by finitely many operations again by exploiting the meaning function μ that one can reach expressions which directly can be decided as 'being the case' or not. In this context it is assumed that a text S is assumed to be *true* because all its expressions are assumed to be *true*. *Being true* means, that the uttering human actors can relate these expressions E_{RW} by their meaning functions to internal neural correlates F_{NN} of the facts which are perceived and learned from the real world facts F_{RW}. Only human actors can manage this kind of *truth*.[4]

Second there is another text which is describing the assumed *dynamics* of the state S which is done by a finite set of *change rules* X (see for change rules figure 3). The general assumption behind the concept of change rules is given in the idea that a state description S represents a certain finite *time slot* T_{Δ} in an assumed time model T and that parts of the state S can *change* thus that a *new state* S' occurs. To that extend one can *imagine* such *possible* changes and one can write down these imagined changes as *change rules* X one is able to have a vague *look into some possible future state*. Considering this case of possible changes one has to determine the possible *causes of change*. Besides those cases where one *does not know* what are the causes for observable changes it is assumed in this text that *every identified actor* can be a possible cause for change.

2 Actors

In figure 3 three main types of actors are distinguished. This list is possibly not complete. If other types of actors will be identified in the future then the list has to be extended.

Everyday Experience: For the following definitions it has to be kept in mind that these definitions assume the perspective of an everyday experience, not

[4]'Real' is not 'real': every neural correlate as part of the inner states IS_{NN} of an actor is the *primary reality* experienced by an actor, partially in the format of a *phenomenon in his consciousness*. But because I can *see* a 'white cup' on the table or I can *imagine* a 'white cup' by recall from memory the brain can distinguish between those *real* experiences which can not easily be changed by 'its own' and those, which can. Thus I can change my remembered white cup in many directions 'by myself'. This difference (enriched by some more aspects) allows the brain to distinguish between the *real real* things and the *only real* things. In the realm of the *real real* things there is another difference between those real real things which are caused from the *body* which contains the brain and the realm *external to the body*, which usually is called the *real world [RW]*. Real real experiences located in the real world can be perceived by more than one actor at the same time. Real real experiences from inside the body – like different kinds of pain caused by the body – can only be experienced by the brain which is inside this body. Despite the differences with regard to a public observability of real real experiences - inside the body, external to the body – the body is generally also part of the real world, because a body is visible to others and the inner parts of the body can be investigated as a real world object.

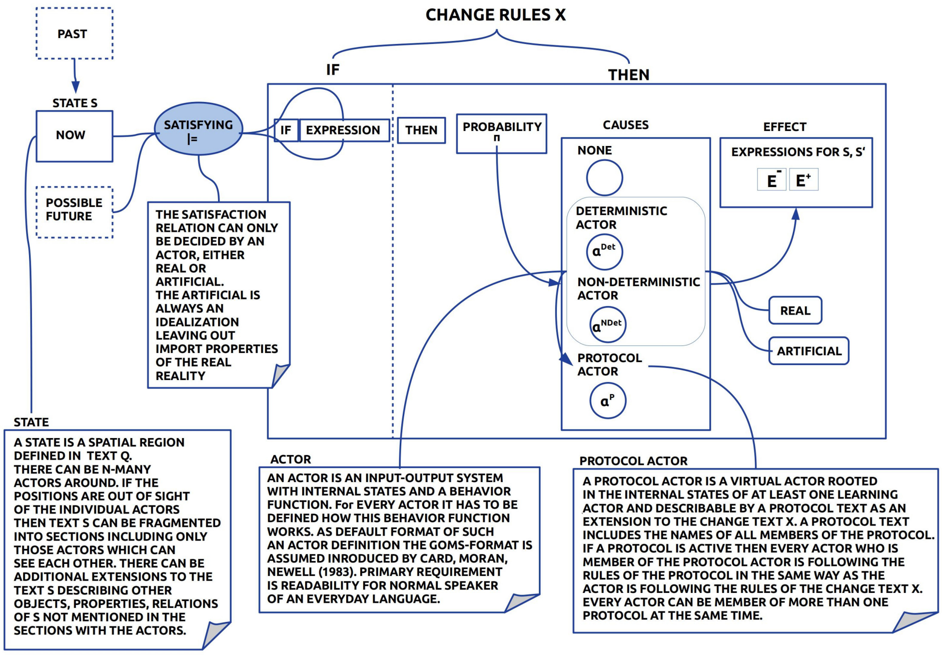

Figure 3: Outline of interplay between text S, text X with explanations of the interacting elements

general science. As example consider the case of *determinism*. In an everyday experience with limited time frames you can experience many events associated with objects (actors) which appear to be deterministic because during short time frames things appear and respond seemingly in *the same way*. Looking to the material of our world with the *eyes of science* we detect that all material is in some way *changing through the course of time*. Thus even buildings constructed out of concrete and steel show phenomena of erosion which destabilize these buildings (especially bridges) which can lead to broken bridges after some time. Thus if in this text *deterministic* actors will be defined then this is related to everyday experience. The same holds for the other definitions.

Deterministic Actor: A *deterministic actor* α^{Det} is assumed to be an input-output system with a *behavior function* ϕ_{Det} which determines which kind of *output* O will occur if a certain kind of *input* I happens, written as

$$\phi_{Det} \quad : \quad I \longmapsto O \tag{1}$$

If one knows the behavior function ϕ_{Det} then one can predict the output of the system if one knows which kind of input is actually given.

Non-Deterministic Actor: A *non-deterministic actor* α^{NDet} is like a deterministic actor but with a difference: the output is not only depending from the input but also from the actual *internal states* IS. Thus the behavior function ϕ_{NDet} is written as

$$\phi_{NDet} \quad : \quad I \times IS \longmapsto O \tag{2}$$

Because one usually can not exactly know which internal states IS are actually active it is not possible to predict the output of the system clearly.

Learning Actor: A sub-case of non-deterministic actors are the *learning actors*. These non-deterministic actors can change their internal states in the light of their experience, written as:

$$\phi_{NLDet} \quad : \quad I \times IS \longmapsto IS \times O \tag{3}$$

$$\phi_{NLDet} \quad \subset \quad \phi_{NDet} \tag{4}$$

$$\phi_L \quad = \quad \phi_{NLDet} \tag{5}$$

In this text the following minimalistic characterization of a learning actor is assumed: if an actor α^{NDet} shows after some point of time t in front of a situation s a behavior which is different to the behavior shown in front of this situation s before t and this new behavior is kept 'for some time interval (t,t')' then the non-deterministic actor α^{NDet} is assumed to be a *learning actor* α^L. From this definition you can not infer anything about the real structures inside the actor. It is the task of the scientist to define hypothetical formal models of the inner structures which are capable to describe the observable behavior of a learning actor α^L. The same observable behavior allows *many different possible formal models*. Nevertheless, as long as the *meaning (the observable behavior)* of the different formal models is the *same* these formal models are *equivalent*.

All kinds of biological system have this learning ability. That it is possible to predict the behavior of biological systems nevertheless to some extend is due to the fact that biological systems are using their learning ability not because they have the 'goal' to be 'different' but to *optimize* the behavior for certain *goals* important for the existence of the system. Such a *goal-directed behavior* can be quite stable as long as the goals are stable and the result of this behavior for the actors is 'as expected'. *Goals* are arbitrary internal states existing for some time, which can change, can disappear or can arise again. If the actor is *exploring* new goals the behavior may appear for an external observer *chaotic*. Besides this a learning actor can have more than only one goal at the same time. No fixed rule is know which manages a set of goals in a learning actor.

Protocol Actor: A social type of actor is the *protocol actor* α^P. A protocol actor is a *virtual actor* consisting of at least one learning actor which has *agreed* to follow a certain *protocol* P for some time T_{Delta}. A protocol is a text with a finite list of change rules which have to be served in the linear order of the protocol[5]. Every individual learning actor which has agreed to follow a protocol P will follow the rules of the protocol. Thus, even if different actors $\{\alpha_1^L, ..., \alpha_n^L\}^P$ which are constituting a protocol actor, are distributed in the state S at different locations without perceiving each other these different actors can and will act according to the common protocol and thereby realizing in a distributed synchronous way a virtual protocol actor with real actions. The *behavior function* of a protocol actor can be understood as:

$$\phi_P \quad : \quad 2^A \times P \times S \longmapsto 2^{ACT_A \times EF} \tag{6}$$

Thus some learning actors $A^L \subseteq 2^A$, knowing an agreed protocol $p \in P$ and being part of the state S will do some action $a \in ACT$ and thereby they will cause some *effect* EF which can change the state S.

2.1 Examples for S and X

Here first simple examples to support an understanding of the before introduced concepts.

State S: *Mary stands before the library of the university*. This expression is assumed to describe some *fact* in the *real world*, which can be decided by an *observer* as *being the case* (= being true) or not (=being false). In this text it is required that a finite set of expressions constituting the *description of a state* S should all be *true*. To write alternatively the expression 'Mary *sees* the library of the university' would only be valid if the observer has interacted with the person Mary and has asked her, whether she can see the university, because from the outside it is not decidable what the internal states of the actor 'Mary' are. The same would be true if someone would write the expression 'Mary will *lend* a book from the library of the university'. Whether the actor 'Mary' will indeed 'lend' a book is not observable from the outside.

[5]The assumption here to associate a *protocol* P with a text is an *idealization*. In the everyday experience human actors are adhering to protocols without using an explicit text. Human actors are learning many different conventions through interaction with their environment, especially with other human actors.

Details for a State S: If a state S presupposes a *spatial region* SR which has some rich details – like a *house*, a complete *street with houses*, a whole *city* or even more – then it would be awesome to write down every time all the details of S. In this case one can prepare a description of parts of the state S in separate texts with *identifiers*. Then it suffices to mention only the identifiers as part of the state S and only if needed one can look to these additional information.

Sections for S: To improve readability for human readers it can be helpful to structure a text for S along the occurrences of actors in S dividing S according to those groups of actors which can perceive each other. Because actors usually can move around the sectioning of S can change.

Deciding Satisfiability: In the general case the ability to decide whether some set of conditions Φ of a change rule are *satisfiable* by S – written as $S \models \Phi$ – is rooted in a *human actor* HA with the appropriate meaning function μ. Nevertheless it can be of great help to construct many different *artificial actors* AA with more simplified and idealized knowledge and meaning functions to explore the combinatorial spaces which are induced by the dynamic of these simplified actors. Especially if one compares the unknown human knowledge space with the known artificial spaces this can shed some light to improve the understanding of the human dynamics.

Change Rules X. Actor Free: The change rules can have a different format depending from the *circumstances*. In the most simple *actor-free case* a rule $\xi \in X$ announces a set of conditions which have to be fulfilled by the actual state S, and if this is the case then with a probability $\pi \in [0,1]$ a set of expressions $E^{-} \subseteq S$ will be removed from the actual state S and another set of expressions $E^{+} \cap S = \emptyset$ will be added to S. The set of expressions $E^{-} \cup E^{+}$ is called the *effect* EF of the change rule.

Change Rules X. With Actors: In an *actor case* one has to distinguish two cases: if there exists a *real human actor* HA_{RW} then after the satisfaction of the conditions with a certain probability the human actor will react to the actual situation S with an *expected action* act_{alpha} associated with an expected effect EF. But a real human actor is free to do something else. In case of an *artificial actor*AA one has to define an *operational definition* of the artificial actor in a way that a *computation* of the output O in terms of an *effect* EF is possible. For this computation the actual state S will be assumed as input I and – in case of a learning actor – additionally the internal states IS

as working factors. For an artificial actor too one can announce an expected action act_{alpha} associated with an expected effect EF. Again, if the artificial actor is a *learning* actor then the learning actor is also free to do something else.

Change Rules X. With Protocol Actors: Compared with the *normal actor case* the only difference is that the actors associated with a protocol text P will for their computation of an output take also into account what the protocol is stating. In an advanced case it is possible that the learning actors associated with a protocol can *change the protocol*!

3 Application Scenario

Having said all this about the theory it has to be clarified how it is possible to use this theory in a social process called here *actor story* AS driven by *human actors* HA which want to share their *experience* $\{F_{NN}, E_{NN}, \mu_{NN}, ...\}$ triggered by some given problem P for to know, how a *possible future* S' of the *actual situation* S will look like. Although it is clear from the stated assumptions about everyday experience that every kind of formalization is an *idealization* and thereby a *simplification* and it should be clear, that all learning actors α^L can behave different from what is *expected* from the *history so far*, the sharing of experience in the above described format can nevertheless shed more light on our actual situation S and possible future(s) S' than doing nothing. Staying in isolation or by limiting oneself with occasional small talks is of no real help. Otherwise sharing experience and doing some common structuring with the above described method can lead to some insights, but insights as such do not change the word automatically. The shared knowledge has to be associated with those decisions in our world which have an effect for the real course of our real world. But history teaches us that improved insights into the logic of the world will not automatically lead to a more improved world. There are – until now – always enough human actors around which have special goals of their own which are not compatible with the goals of shared experience of the everyday experts.

To enable the above vision of the construction of an appropriate actor story AS the following application scenario is assumed:

1. There exist a *group of experts* Γ whereby every human actor is assumed to be an expert.
2. The group of experts has agreed to a given *problem* P, which can be described in a shared text D_P.
3. The described problem can be associated with some *spatial region of the real world* SR and with a *time model* containing a *time frame* T_Δ structured by *cycles* CYC.

4. There exists an assumption *which kinds of actors* shall be involved in the problem, where and how they are *initially located* in the spatial region.

5. The experts will produce a text D_S describing an *actual state* S as part of the problem. A *state* S is assumed to be a collection of *facts* F which are assumed to be the case in a certain given short time frame T_0 and this assumption can be *decided directly* by every member of the group of experts Γ.

6. The experts will produce another text D_X containing *change rules* X which describe which kinds of changes are assumed to be *possible* with regard to the *given actual state* S or which are acknowledged to be *helpful* to *improve* the actual state S.

7. The experts have also agreed to a mechanism $\models$ how a human actor is able to decide whether a given change rule $\xi \in X$ can be said to be *satisfied/ fulfilled* by a given actual state S, written as $S \models \xi$. At the same time there can exist more than only one change rule which can be satisfied by an actual state S, this is the subset of *satisfiable change rules* $X^{\models,S} \subseteq X$ in state S. Substituting *real human actors* HA_{RW} by *artificial actors* AA is principally possible but this implies an *idealization* and thereby a *simplification* which will not represent the full potential of the real world as such. Nevertheless it can be of help for the human experts to understand better in some parts the complexity of everyday experience.

8. The two texts S and X should be given in a format which allows to apply the change rules X to the state description S in a way that it can be decided what is a follow-up state S' of the actual state S. The follow-up state S' then will become the new actual state S and the change rules can be applied again to S. This process of applying the change rules X to the actual state description S will be repeated until there is some *stop criterion* which can be decided to be fulfilled.

9. Such a process of a repeated application of the change rules to the state description is in this text called a *simulation*.

10. For a simulation to happen two cases are assumed: (i) **Without a computer:** Human actors apply the change rules X to the actual state S in the light of their usual understanding. (ii) **With a computer:** The computer supports the (ii.1) editing of the texts D_S, D_X as well as (ii.2) possible formal descriptions of *artificial actors* as well as (ii.3) a formal version of the *satisfaction relation* $\models$ as well as (ii.iv) a complete simulation. But even in the case where a computer supports (ii.1) - (ii.iv) it will be possible that human actors are participating in the simulation as real actors.

4 Actor Story [AS]

The *actor story* AS translates the application scenario into a sequence of concrete states which describe the *interactions* of the *intended users* of the system – the *executive actors* α_{exec} – with the *assisting system* α_{ass}. That a user can interact with the system it has to be assumed that there exists a *system interface* SI which allows the user to make some *inputs* to the system as well as to receive some *outputs* from the system. Strictly speaking the user itself has also an interface called the *user interface* UI and which kind of actions the user is capable of or which kinds of inputs the user is capable to receive depends from this user interface.[6]

4.1 Start

In the start state it is assumed that there is at least one user before the system and the system interface SI invites to start the process.

Task: Start the process.

Actors: Human experts.

SI: A main window W1 showing all possible options:

(a) *Edit P* (the problem document D_P).
(b) *Edit S and Edit X* in parallel.
(c) *Simulate* (Apply X to S).
(d) *Evaluate* the whole process after the simulation has finished.
(e) *Stop* the simulation.

Actions: Select an option.

4.2 Editing P

Task: Input all data which are necessary for the problem document D_P.

Actors: Human experts.

SI: A main window W1 with a menu showing all possible questions to be answered.

(a) Describe the *problem* P

[6] Because people can have great differences in their user interfaces as well as in their cognitive capabilities a full analysis of the user-system interaction had to address all these cases. In the context of the actual project we have still a strongly experimental setting and it will be considered only the case of no special handicaps.

(b) Describe the intended real part of the world (*space*).

(c) Describe the *time* model T: which time period, which cycles.

(d) Which *actors* A are participating in the scenario.

(e) Some other assumptions.

Actions: Select every question and write an answer.

4.3 Editing S and X State in Parallel

This state allows the editing of the texts S and X in parallel, but one must not. Additionally one can call from within this state the simulation mode to test whether the actual texts are working.

Task: Input all data which are necessary for the S-state (including sectioning and extended texts with details)

Actors: Human experts.

SI: A main window W1 offering the editing of a text consisting of individual statements. Every statement can be edited separately and repeatedly.

Actions: Select either a given statement for editing or edit a new statement or stop.

Task: Input all data which are necessary for the X-state (including different protocols for protocol actors). If actors will be used in the state S then the behavior functions of these actors have to be specified.

Actors: Human experts.

SI: A main window W1 offering the editing of a text consisting of individual statements. Every statement can be edited separately and repeatedly. Every statement has the format 'IF ... THEN ...' according to the theory.

Actions: Select either a given statement for editing or edit a new statement or test the simulation or stop.

4.4 Simulation

General: The simulation mode depends highly from the kinds of actors which are involved. In a *human only* simulation all decisions will be made by human actors. In an *artificial actor only* simulation the whole simulation can be done completely automatically. In a *mixed simulation* real humans as well as artificial actors can interact.

Task: The Program which manages the simulation is called a *simulator* σ. According to the situation it has to manage the application of the change rules X onto the actual state S. The simulator σ computes a series of states starting with the state S_0. The simulation will stop according to an agreed *stop criterion*.

Actors: Human actors as well as – optionally – artificial actors

SI: After starting the simulation one sees two windows: W1 shows the actual state and W2 shows the rules which will be applied.

Actions: The simulator computes a new state S' by applying the change rules X. The follow-up state S' then becomes the new actual state S. The process can be repeated. If the simulator is unable to determine whether a certain change rule $\xi \in X$ can be applied to the actual state S then the simulator asks the human experts for a judgment.

4.5 First Simulator Cycle

As a first simple simulator cycle the following schema has been proposed (cf. figure 4):

1. If there already exists a state file S_i and a rule file X_i load such a file, otherwise edit two new files.
2. The simulator σ works in *cycles*.
3. Every cycle CYC_i the actual version of a state description S_i as well a rule set X_i will be loaded into the simulator.
4. If the set of applicable *Conditions* $Cond^* \subseteq Cond$ is *empty* $\emptyset$ then the user can edit new rules R_{new} and can add these to the existing rule set R_i thus extending the rule set: $R_{i+1} = R_i \cup R_{new}$.
5. If the set of applicable rules is not empty $\neg\emptyset$ then the user can nevertheless add new rules or he can apply the fitting rules.
6. Applying change rules X^* to the actual stae S_i will change the state according to the schema: $S_{i+1} = S_i - E^- \cup E^+$.
7. After the simulation has finished both original sets S_i and X_i have been change – X_i not necessarily. They can be stored permanently.

4.6 Evaluation

Task: After a simulation the experts have the possibility to analyze the simulated process by different criteria.

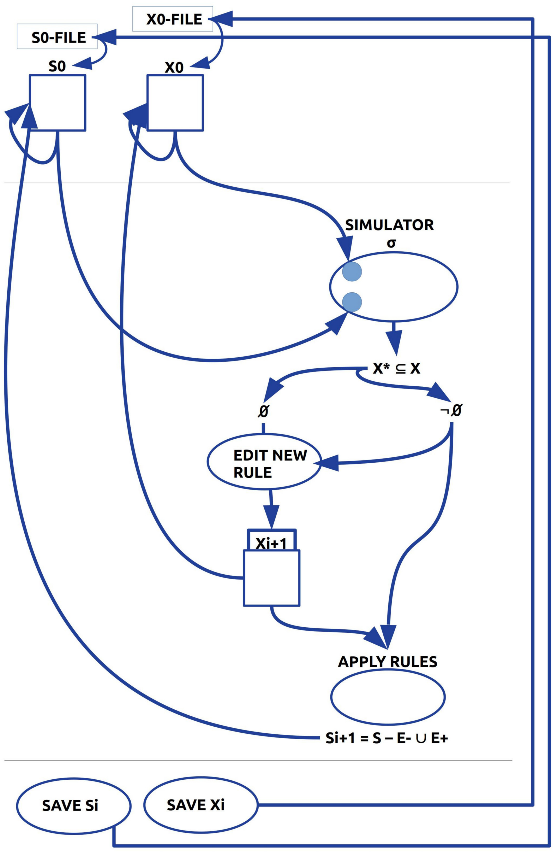

Figure 4: First simple simulator cycle

Actors: Human experts.

SI: After starting the simulation one sees two windows: W1 shows the possible criteria which can become activated for an evaluation and W2 shows the results with regard to the criteria.

Actions: The human experts select those criteria which should be commented by the system and read then the output.

4.7 Stop

Task: End the process.

Actors: Human experts.

SI: Bye Bye window

Actions: Quit.