Web Mobility

Web computing between semiotic and kenomic spaces

Rudolf Kaehr Dr.@

ThinkArt Lab Glasgow

Abstract

Locality, positionality and mobility in semiotic, categorical, diamond and kenomic systems. Kenomic mobility compared with Agha's Universal Actor System (UAM) and Middleware approach and Milner's Bigraphs. Sketch of an Architectonics of Kenomic Mobility. Introducing trans- and diamond-Actors and their chiastic interplay as interactional and reflectional actors in knowledge grids.

Web mobility as we know it is based on a mono-contextural static organizational system, mobility then is restricted by its static numerical framework. Kenomic mobility starts to diamondize such a static framework towards a metamorphic dynamics of polycontextural diamond structurations.

1

1. Kenomic mobility

1.1. Monomorphy

Mobility based on semiotics, i.e. sign systems, is restricted by the semiotic *equality* rule. Existence (occurrence) and locality of signs are identified in sign systems. There is no reason to separate the *identity* of an atomic sign from its *locality* in a sign sequence (word). Both notions, "occurrence" and "locality" of signs are coinciding. An atomic sign might occur as a graphemically (or:syntactically) identical sign, say "a", at different places in a sign sequence (string, word) but its graphemic identity as "a" is independent of the place (locus, position) it occurs. The identity of the sign is pre-given to semiotics and is has its tectonic place in the sign repertoire (alphabet) of the sign system. The rules (economy) of the sign system is not involved into the definition of its signs. This is the abstractness of sign systems, codified by Markov's axiom of the *"Abstraction of Identification"*.
It would be crazy if a sign would change its graphemic identity in regard to its position (occurrence) in a sign sequence. Potential identification and potential iterability of signs goes hand in hand.

An example which is working constructively with the notions of position and sign is the *positionality* system for natural numbers. In a positional system the *value* of a graphemically identical cipher is changing in respect to its position. The number "1", is of different value if it is at position one, i.e. "1", or at position one, two and three, of the positional string, say "111". But it would be utter nonsense, if the cipher "1" at position one would have the form of the cipher "3" and at position three the form of cipher "8". The result would be the number "138" and not "111".

Another, although less known use of the idea of positionality, is given by Gotthard Gunther's *place-valued* systems for the distribution and mediation of logical systems which culminates with the concept of polycontextural logic. A further development was introduced by the dissemination of natural number series based on the place-designator for number systems. In several texts I introduced the concept of a *kenomic matrix* for the dissemination of formal systems in general and their interactionality, reflectionality and interventionality.[2]

Nevertheless it seems that exactly this craziness of a position-dependent identity of graphemic objects might be the next step in the deliberation of scriptural design from its inherent semiotic limitations.

Signs might have different meanings, i.e. polysemy, or a single meaning might have different sign representations, or it might even be monosemic: one sign, one meaning. But all the possible cases are based on the distinction of sign and meaning (or value, etc.).

This is the field of semiotic and logical thinking. Here, sign systems are conceived as the medium (or even instrument, tool) of thinking. Hence the use of signs in sign systems is not changing the identity of its signs.

A graphmatic turn

The *graphematic turn* is focusing on the difference of sign and position. It is thematizing both together, the notion of difference and the dichotomy of sign and position. In this sense, the *blind spot* of semiotics is its blindness for the co-creative interplay of locus and mark (sign, object). Locus and mark are positioning the difference of token and type of a sign.

This is the field of *graphematic* and *grammatological* scriptures (adventures, studies).

The idea of a separation of marks and loci is not absolutely new. Similar ideas have had some occurrences at several places in the context of kenogrammatics and in an interpretation of George Spencer Brown's *Calculus of Indication*. For the understanding of the Calculus of Indication the idea of a *"topologically invariant"* notation is mentioned by Matzka[3] .

"Obviously, a kenogram is composed of some sort of "atoms", in the sense of indivisible parts, but those "atoms" have no identity as types. In fact, if we ask how many "atoms" there are, and if we equate "atom" with "kenogram of length one", then the answer is that there is one and only one "atom". The concept of an alphabet, as a set of two or more types of atoms, becomes obsolete in the context of kenograms. Because of this very strange property of the kenogrammatic "atoms", we term them "kenoms", so that the kenograms can be called "strings of kenoms". (Matzka 1993)

Topological invariant notational systems, like the Calculus of Indication, are abstracting from the locus a mark takes place, but they are not yet studying the *interaction* between locus and identity (occurence) of marks or signs. Marks are occuring as single atomic elements, there is no concept of patterns of marks involved.
The argument related to the lack of atomicity in kenomic systems is not taking into account the genuine kenomic structures (patterns) of *monomorphies*. Monomorphic patterns (monomorphies) are basic in kenomic systems.

Despite the fact that "strings of kenoms", i.e. morphograms, consisting of kenograms, can be build recursively by the "successor" operations of *iteration* and *accretion*, the decomposition of morphograms is not a reduction to atomic signs or even to one and only one atomic sign.

From the point of view of a monorphic decomposition, an atomic sign is a *monadic* monomorphy and not a semiotic atom. There are no atomic signs in kenogrammatics, simply because their are no signs at all involved in kenogrammatics.
The basic "elements" of kenogrammatics are morphograms consisting of monomorphies and the "content" of monomorphies consists of kenograms.
[Or in the terminology of Matzka, kenograms consist of kenoms, building "strings of kenoms", called kenograms.]

Lack of a pre-given alphabet

As a surpising result we get the fact that there is no alphabet in kenomic systems, keno- and morphogrammatics. Atomic signs as members of the set "alphabet" don't exist. Each "atomic kenom" is kenomically equivalent. Further more, we can state, there is no alphabet as the beginning of all words, the morphograms themselves are the alphabet without any beginning. This intriguing phenomenon is studied in extenso in my eBook "Skizze 0.9.5".

"So where is the Chinese alphabet[4]*and why is it so hard to find on the web? Well, the main reason is that there is no such thing as an alphabet in China."*

Because of the lack of an alphabet as a source for signs from the outside, i.e. from a lower level of the tectonics of a morphogrammatic calculus, evolution of morphograms have to be constructed as extensions out of their inner structure. This is a kind of an *immanent* evolution of morphograms based on the mononomorphies of the morphogram.

Self-generated alphabets

The wording that there is *no* alphabet means, there is no alphabet pre-given as the start of a kenogrammatic calculus. But what's not pre-given is not denied to exist in a different way. Hence, a positive wording concerning the alphabet of kenogrammatics might be turned into this: Encountered a morphopgram, a kenomic abstraction is collecting the kenoms involved into the morphogram. A successor operation then can rely on those kenograms to precede to the next morphogram, in an iterative or an accretive way.
Therefore, albeit there is no alphabet pre-given, kenogrammatic operations are producing situatively their own alphabet, i.e. set of kenoms, to proceed their operations.
Again, it is reasonable to speak about a parallelism or diamond movement of operators and operands of kenomic operations. The kenomic alphabet has to be elicited. There is no need for

a kenomic alphabet without intended interactions with morphograms.

Paradox of inscription

There is surely an additional paradox involved in writing morphograms. Until now, morphograms have to be written by signs. Hence, there seems to be a semiotic dependence for morphograms. Without signs, there are no morphograms.

This is true, as much as it is true, that there are no signs without physical marks. Hence, semiotic signs are depending on physical matter. And thus, there is no semiotics without physics. Again, this is a circular argumentation. To draw the distinction of signs and matter, signs have to be used. Morphograms are using signs but they are not signs. The scriptural media are enlarged to: marks - signs - morphograms. Between sign systems and morphogrammatics, a new interactivity is opened up. The interaction between matter (marks) and signs is based on a graphical level (typography) and is not yet reaching the intelligible level of sign systems.

Following the terminology of Gunther, morphograms are involved into *evolution* and *emanation*. Evolution happens with *iterative* and *accretive* successions (disremptions). Emanation with *differentiation* and *reduction* of morphograms.

Thus, monomorphic decomposition, as well as monomorphic composition, is a different topic of kenogrammatics and shouldn't be confused neither with the mentioned operations of kenogrammatics nor with similar semiotic operations.

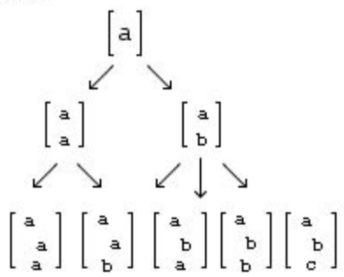
Monomorphic decomposition

Morphograms are decomposable, not into atomic signs, called kenoms but into kenomic patterns, called *monomorphies*. Monomorphies of decomposed morphograms are collected as ordered sets, i.e. n-tuples, and not as sets only. The order of the components is preserving the structure of the steps of decomposition.

Example for $MG^{(3)}$:

Composition of Morphograms

$\mathbf{MG}_{Trito}{}^{(3)} =$

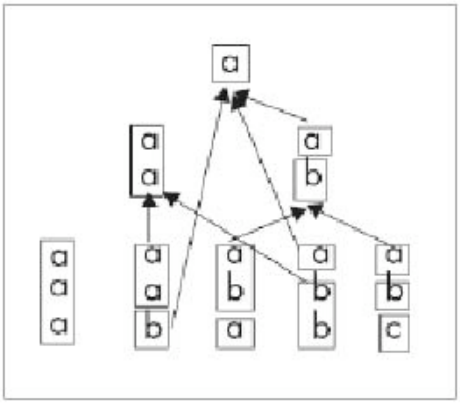


$MG^{(3)} = \{[aaa], [aab], [aba], [abb], [abc]\}.$

Decomposition of the morphograms of $MG^{(3)}$ into tupels of monomorphies.

$\mathbf{Dec}(\mathbf{MG}^{(3)}) =$

1. $[aaa] \longrightarrow ([aaa])$
2. $[aab] \longrightarrow ([aa], [b])$
3. $[aba] \longrightarrow ([a], [ba]) \longrightarrow ([a], [b], [a])$
4. $[abb] \longrightarrow ([a], [bb])$
5. $[abc] \longrightarrow ([ab], [c]]) \longrightarrow ([a], [b], [c]).$



Dec(MG$^{(3)}$) ={([aaa]), ([aa], [b]), ([a], [b], [a]), ([a], [bb]), ([a], [b], [c]}

Explanation:
1. Dec:[aaa] --> ([aaa]) :
Also [aaa] is constructed by the steps [a]-->[aa]-->[aaa], a decomposition into monomorphies is not accessible because the sole monomorphy, i.e kenomic pattern, is [aaa] itself.
2. [aab] --> ([aa], [b]):
The morphogram [aab] is decomposable into its monomorphies [aa] and [b]. Dec([aa]) = [aa].
3. Dec:[aba] --> ([a], [ba]) --> ([a], [b], [a]) :
How are the monads differentiated? Obviously not by their monadicity only. But also by their *position* in the kenomic pattern as it is the case for the decomposition of [abc] into its monads ([a],[b],[c]), hence monads [a] and [b] of the decomposition of the morphogram [aba] has be accepted as different monads; and thus the decomposition of [aba] into the 3-tuple ([a], [b], [a]) gets a legitimatimation.

Both decompositions ([ab]), [a]) and ([a], [ba]) are resulting in a further step of decompsition into ([a],[b],[a]).

```
Dec([aba]) = (Dec([ab]), [a]) = ([a],[b],[a]),
Dec([aba]) = ([a], Dec([ba])) = ([a],[b],[a]).
Dec:([aba]) -->([ab], [a]) -->([a],[b],[a]),
Dec:([aba])  -->([a], [ba]) -->([a],[b],[a]).
```

4. Dec:[abb] --> ([a], [bb]):
The morphogram [abb] is decomposable into the monomorphies [a] and [bb]. Dec([bb]) = [bb].
5. Dec:[abc] -->([ab], [c]])-->([a], [b], [c]):
The morphogram [abc] is decomposable step-wise into its monadic monomorphies.
Both decomposition steps

```
Dec:[abc] -->([a], Dec([bc]]))-->([a], [b], [c]) and
Dec:[abc] -->(Dec([ab]), [c]])-->([a], [b], [c]) are equivalent.
```

To choose the first in a representation is thus a question of convention in respect to the lexical order of the marks.

Comments:
The decompositions for [aab] and [abb] are delivering the same *set* of monomorphies, i.e. one monad and one dyad. Because the position of monomorphies in a morphogram is of relevance for morphograms of the trito-level of kenogrammatics (in contrast to the proto- and deutero-level), the order of the decomposition has to be taken into account. This is easily shown with the *reflection* of the morphograms: R([aab])= [abb]. Hence the order has to be kept and decompositions are producing *tuples* and not sets of monomorphies. This holds specially for the reflection of the morphogram [aba]: R([aba]) = [aba].
For morphograms of the deutero- and proto-level the order can be omitted and the to morphograms, [aab] and [abb], would be equivalent.
A full-fledged study of morphogrammatic systems, based on a “concatenational” approach, is available as Morphogrammatik . *Eine Einführung in die logische Theorie der Form.*[5] (Mahler, Kaehr, 1993) A new more pattern-oriented approach will be published soon as an *"Outline of Morphogrammatics".*

Reflection in MG $^{(3)}$:
Further insight into the order of monomorphies in morphograms can be achieved with the simple operation of *reversion*. It turns out that the morphogram [abb] is the reverse of morphogram [aab] and not its equal.

$$R([aaa]) =_{MG} [aaa]$$
$$R([aab]) =_{MG} [abb]$$
$$R([aba]) =_{MG} [aba]$$
$$R([abb]) =_{MG} [aab]$$
$$R([abc]) =_{MG} [abc].$$

Decomposition and reflection are interchangeable: R(Dec(MG))=Dec(R(MG)).

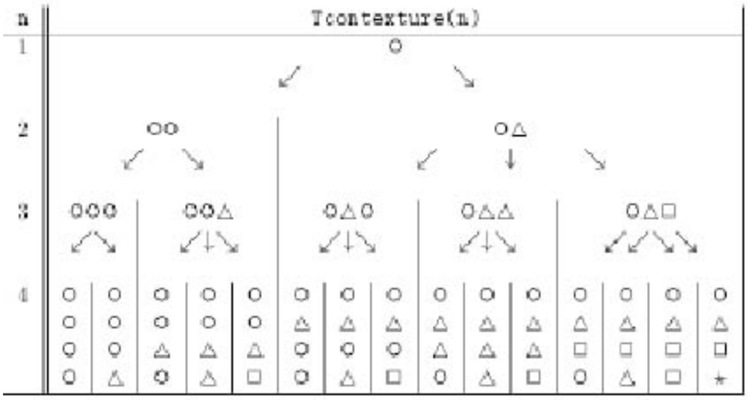
```
1.  R(Dec([aaa])=R([aaa])=[aaa]
    Dec(R([aaa])=Dec([aaa])=[aaa]
2.  R(Dec([aab])=R([aa],[b])=([b],[aa]) = [abb]
    Dec(R([aab])=Dec([baa])=([b],[aa])
3.  R(Dec([aba])=R([a],[b],[a])=([a],[b],[a])  = [aba]
    Dec(R([aba]))=Dec([aba])=([a],[b],[a])
4.  R(Dec([abb]))=R([a],[bb])=([bb],[a]) = [aab]
    Dec(R([abb])=Dec([bba])=([bb],[a])
5.  R(Dec([abc]))=R([a],[b],[c])=([c],[b],[a]) = [abc]
    Dec(R([abc])=Dec([cba])=([c],[b],[a]).
```

The reflector R is defining a simple structure on the morphogrammatic system $MG^{(3)}$:

$[MG^{(3)},R]$ with $[Mg_1]\to[Mg_1], [Mg_2]\longleftrightarrow[Mg_4], [Mg_3]\to[Mg_3], [Mg_5]\to[Mg_5]$,

$[MG^{(3)}, R] = ([Mg_1], [Mg_2]\longleftrightarrow[Mg_4], [Mg_3], [Mg_5])$.

Composition and decomposition for $MG^{(4)}$



$MG^{1} = [aaaa]$, $MG^{2} = [aaab]$, $MG^{3} = [aaba]$, $MG^{4} = [aabb]$

$MG^{5} = [aabc]$, $MG^{6} = [abaa]$, $MG^{7} = [abab]$, $MG^{8} = [abac]$

$MG^{9} = [abba]$, $MG^{10} = [abbb]$, $MG^{11} = [abbc]$,

$MG^{12} = [abca]$, $MG^{13} = [abcb]$, $MG^{14} = [abcc]$, $MG^{15} = [abcd]$

Decomposition of the 15 morphograms of MG $^{(4)}$:

1. Dec ([aaaa]) = [aaaa]
2. Dec ([aaab]) = ([aaa], [b])
3. Dec ([aaba]) = ([aa], [ba]) ⟶ ([aa], [b], [a])
4. Dec ([aabb]) = ([aa], [bb])
5. Dec ([aabc]) = ([aa], [bc]) ⟶ ([aa], [b], [c])
6. Dec ([abaa]) = ([ab], [aa]) ⟶ ([a], [b], [aa])
7. Dec ([abab]) = ([ab], [ab]) ⟶ ([a], [b], [a], [b])
8. Dec ([abac]) = ([aba], [c]) ⟶ ([a], [b], [a], [c])
9. Dec ([abba]) = ([abb], [a]) ⟶ ([a], [bb], [a])
10. Dec ([abbb]) = ([a], [bbb])
11. Dec ([abbc]) = ([abb], [c]) ⟶ ([a], [bb], [c])
12. Dec ([abac]) = ([aba], [c]) ⟶ ([a], [b], [a], [c])
13. Dec ([abcb]) = ([ab], [c], [b])
14. Dec ([abcc]) = ([ab], [cc]) ⟶ ([a], [b], [cc])
15. Dec ([abcd]) = ([abc], [d]) ⟶ ([a], [b], [c], [d])

How to construct monomorphies mathematically?

From a mathematical point of view, monomorphies are *partitions* of mappings. This is well elaborated by [Schadach 1967]. The procedure to build monomorphies out from morphograms, as it is mathematically defined by Schadach's approach, shall be called *monomorphic decomposition*, short “Dec”. Hence, Dec(MG) is the operation to produce monomorphies from morphograms MG.

" Let A and B be non – empty finite sets,
$A = \{a_1, a_2, \ldots, a_n\}$, $B = \{b_1, b_2, \ldots, b_m\}$. Let denote B^A the set of all mappings from A to
$B^A = \{\mu \mid \mu : A \longrightarrow B\}$, $\text{card}\, B^A = (\text{card}\, B)^{\text{card}\, A} = m^n$.
The following theorem shows that
every family of subsets B^A defines a certain *partition* of B^A.

Theorem 1.
Let $\{R_i \mid i \in I\}$ be a family of subsets of B^A where I is a finite index set; $R_i \subseteq B^A$ for each i ∊

The family $\{R_i \mid i \in I\}$ defines a partition of B^A such that
the elements of the partition (the equivalence classes of mappings) are

$$[\mu]_{I_x} = \bigcap_{i_x \in I_x} R_{i_x} - \bigcup_{i_y \in I - I_y} R_{i_y}$$

where I_x runs through all subsets of I.

Corollary 1.

If $I_x = \emptyset$, then $[\mu]_\emptyset = B^A - \bigcup_{i \in I} R_i$ and

if $I_x = I$, then $[\mu]_I = \bigcap_{i \in I} R_i$.

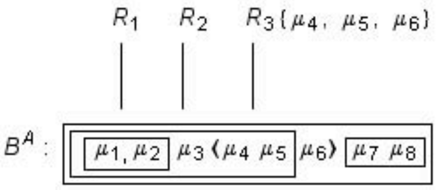
Corollary 2

By Theorem 1, we get a mapping from the set of all families of subsets of B^A onto the set of all partitions of

Example.

Let be $B^A = \{\mu_1, \mu_2 \ldots, \mu_8\}$ and the family of subsets $\{R_i \mid i \in I = \{1, 2, 3\}\}$ where

$R_1 = \{\mu_1, \mu_2\}$,
$R_2 = \{\mu_1, \mu_2, \mu_3, \mu_4, \mu_5\}$,
$R_3 = \{\mu_4, \mu_5, \mu_6\}$.



I_x	$[\mu]_{I_x}$
$\emptyset$	$B^A - (R_1 \cup R_2 \cup R_3) = \{\mu_7, \mu_8\}$
$\{1\}$	$R_1 - (R_2 \cup R_3) = \emptyset$
$\{2\}$	$R_2 - (R_1 \cup R_3) = \{\mu_3\}$
$\{3\}$	$R_3 - (R_1 \cup R_2) = \{\mu_6\}$
$\{1, 2\}$	$(R_1 \cap R_2) - R_3 = \{\mu_1, \mu_2\}$
$\{1, 3\}$	$(R_1 \cap R_3) - R_2 = \{\mu_7, \mu_8\}$
$\{2, 3\}$	$(R_2 \cap R_3) - R_1 = \{\mu_4, \mu_5\}$
I	$R_1 \cap R_2 \cap R_3 = \emptyset$

B^A: [[μ_1, μ_2] [μ_3] [μ_4 μ_5] [μ_6] [μ_7 μ_8]].

(Dieter J. Schadach, BCL Report No. 4.1, August 1, 1967)

Monomorphic mobility

On the base of monomorphies in kenomic systems, identity and locality are separable.
That is, monomorphies at two comparable locations of two equivalent morphograms might be interchangeable despite their semiotic difference.
Or, the locality of monomorphies in morphograms might differ semiotically.
Hence, monomorphies are interchangeable in morphograms.
That is, morphogrammatic dissimilarity is stable under monomorphic exchange.

That doesn't mean that everything is interchangeable with everything.
Rules of pattern-invariance have to be applied.

Palindromes

For semiotic systems, *palindromic* symmetry seems to be the only possibility for semiotic identity to exchange parts and preserving the identity of the word.

$$(a_5\, a_4\, b_3\, a_2\, a_1)_i = \overline{(a_1\, a_2\, b_3\, a_4\, a_5)_j} \cong (aabaa)$$

$$(a_5)_i = (a_1)_j$$

Semiotic substitution

Interchangeability in semiotic systems is known as *substitution*. Substitution of parts in a semiotic sequence (chain, string, word) is correct if only if it fulfills the rules of identity. That is, equality of two equal semiotic sequences H_1 and H_2 is preserved under substitution iff equal parts, k_1, k_2, are substituted, $Subst_{h/k}$, at equal places (parts), h_1, h_2, of the sequences H_1 and H_2 of the semiotic sequences H.

Prerequisits:
1) Decomposability into parts, based on atomic elements,
2) Measurement of the length of sequences, this is given by (1),
3) Identity of signs and sign sequences.

The domain of signs to be involved in the process of substitution is given by the number of signs of the free monoid based on the sign repertoire.

Semiotic Equality:

$Seq_x\ equiv_{sem}\ Seq_y \Longleftrightarrow \forall\, i, j \in Seq_{x,y}, : loc_i = loc_j \bigwedge loc_i(atom_x) = loc_j(atom_y).$

Or:

$Seq_x\ equiv\ Seq_y \Longleftrightarrow \forall\, i, j \in Seq_{x,y},$

$lenght(Seq_x) = lenght(Seq_y) \bigwedge \forall\, i, j \in Seq_{x,y} : loc_i(atom_x) = loc_j(atom_y).$

Short:

$A = (a_1, a_2, \ldots, a_n),\ B = (b_1, b_2, \ldots, b_m)$

$A =_{sem} B$ iff

1) $m = n$

2) $\forall\, i, 1 \leq i \geq m, n : a_i =_{graph} b_i.$

Again, the equality of two words in a semiotic system is established by the graphemic identity (equality) of the signs at the same locality (position) of the compared words.
The fact of the identification of position and identity of signs has a very clear consequence for the equality of two sign sequences (words). Two words of different lenght are semiotically unequal.

Semiotic Substitution:

$$\forall\, h,\ k \in \mathrm{H}:\ \mathrm{H}_1\ \mathrm{equv}\, \mathrm{H}_2 \Longleftrightarrow \mathrm{Subst}_{h/k}\left(\mathrm{H}_1\right) = \mathrm{Subst}_{h/k}\left(\mathrm{H}_2\right)$$

$$k_1 =_{\mathrm{sem}} k_2 \text{ iff } \mathrm{length}\left(k_1\right) = \mathrm{length}\left(k_2\right) \bigwedge \forall\, i,\ j \in k_1,\ k_2 : \mathrm{loc}_i\left(\mathrm{atom}\right) = \mathrm{loc}_j\left(\mathrm{atom}\right)$$

Example

$$\left.\begin{array}{c} \mathrm{H}_1 = \mathrm{H}_2 = \left(\mathrm{aabbcde}\right) \\ h = \left(\mathrm{bb}\right),\ k_1 = k_2 = \left(\overline{\mathrm{lkmbc}}\right) \end{array}\right\} : \mathrm{Subst}_{\mathrm{bb}/\mathrm{klmbc}}\left(\mathrm{aabbcde}\right) = \mathrm{Subst}_{\mathrm{bb}/\mathrm{klmbc}}\left(\mathrm{aabbcde}\right)$$

$$\Longrightarrow \left(\mathrm{aa}\,\overline{\mathrm{lkmbc}}\,\mathrm{bbcde}\right) = \left(\mathrm{aa}\,\overline{\mathrm{lkmbc}}\,\mathrm{bbcde}\right).$$

$$\forall\, h,\ \mathrm{k}_1,\ \mathrm{k}_2 \in \mathrm{H}:\ \mathrm{Subst}_{h/k_1}\left(\mathrm{H}_1\right) = \mathrm{Subst}_{h/k_2}\left(\mathrm{H}_2\right) \Longrightarrow \mathrm{k}_1 \equiv \mathrm{k}_2$$

$$\left(\mathrm{aa}\,\overline{\mathrm{lkmbc}}\,\mathrm{bbcde}\right) = \left(\mathrm{aa}\,\overline{\mathrm{lkmbc}}\,\mathrm{bbcde}\right) \Longrightarrow \overline{\mathrm{lkmbc}} \equiv \overline{\mathrm{lkmbc}}$$

Obviously, a substitution with different substituents results automatically into semiotic non – equality.

$$\left.\begin{array}{c} \mathrm{H}_1 = \mathrm{H}_2 = \left(\mathrm{aabbcde}\right) \\ h = \left(\mathrm{bb}\right),\ k_1 \neq k_2 : k_1 = \left(\overline{\mathrm{lkmbc}}\right),\ k_2 = \left(\overline{\mathrm{klmop}}\right) \end{array}\right\} : \mathrm{Subst}_{\mathrm{bb}/\overline{\mathrm{klmbc}}}\left(\mathrm{aabbcde}\right) \neq \mathrm{Subst}_{\mathrm{bb}/\overline{\mathrm{klmop}}}\left(\mathrm{aabbcde}\right)$$

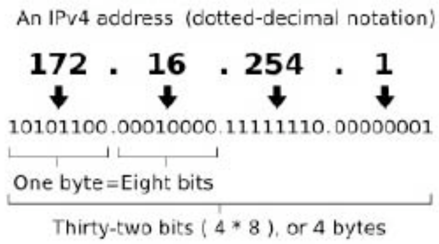
$$\Longrightarrow \left(\mathrm{aa}\,\overline{\mathrm{lkmbc}}\,\mathrm{bbcde}\right) \neq \left(\mathrm{aa}\,\overline{\mathrm{klmop}}\,\mathrm{bbcde}\right).$$

Semiotic substitution happens as abstract as the sign sequences are build by concatenation. There is no need to reflect on the enviroment (context) of the place where the substitution takes place.

Recall: IPs are binary numbers

"An Internet Protocol (IP) address is a numerical identification (logical address) that is assigned to devices participating in a computer network utilizing the Internet Protocol for communication between its nodes. Although IP addresses are stored as *binary numbers*, they are often displayed in more human-readable notations, such as 192.168.100.1 (for IPv4), and 2001:db8:0:1234:0:567:1:1 (for IPv6).

The role of the IP address has been characterized as follows: *"A name indicates what we seek. An address indicates where it is. A route indicates how to get there."* (WiKi)



"A name indicates what *we seek. An address indicates* where *it is. A route indicates* how *to get there."*

The clean identity prerequisites are well collected by the citation:

Name: A name identifies the object. Giving a name, i.e. to address, is to identify the addressed object.

Address: Adressing is to identify the singular location of the object . The process of identifying an object is not changing the identity of the object of identification.

Route: A route is determined by the identities of name and address. Both are not changed in the process of finding the route. There may be many routes between a name and an address but the routing operators are not changing 'en route'.

All three distinctions are defining a *binary relation*: route(name, address).

Such a *"role of the IP address"* transforms directly into the identity characterization of *semiotic substitution*, subst(word, sub).

Applying a morphogrammtic abstraction on binary numbers, binary numbers would be reduced to patterns where the equivalence of (0) ^=(1) holds. As a result, we get, e.g. (10101100) ^= (01010011). Morphogrammatically, the binary numbers (10101100) and (01010011) are represented by the morphogram [ababbaa]. On the other hand, the morphogram might represent numbers from other than binary systems too.

This might be bad news for *digitalism*. Based on the strict distinction of two basic elements “0” and “1”, a morphogrammatic abstraction would undermine not only the possibility of IP addresses but also their calculation on computer systems based on binarism. But *“zero-and-one”* rationality and technology is not more than the tip of the ice-berg of what is possible to realize in a post-digital age of computation.

Monomorphic substitution

What follows is part of a study, which shall be called “*Sign systems in morphogrammatics*”. Topics are the *interactions* between semiotics and morphogrammatics on different tectonic levels of the semiotic and the kenogrammatic systems. This title is emphasizing the fact, that semiotics (word arithmetic, string theory) remains untouched in this exercise. There is nothing wrong with semiotics. It simply has to be disseminated. Complementary, morphogrammatics has to be introduced (constructed) by an interplay with semiotics.

Morphogrammatics offers the possibility of a concept of monomorphic substitution which is a generalization of the semiotic concept of substitution. Monomorphic substitution is not depending on the semiotic equivalence of the substitutional parts, substitutes, but on morphorgrammatic equivalence only. Because morphograms are not abstract sign sequences but wholes, a substitution of monomorphies by other monomorphies which might be semiotically different, the part/whole reationship of the morphogram has to be considered by the substitution process. That is, the monomorphy structure of the morphogram has to be preserved under the interaction of semiotic substitution.

In the following, equal length of morphograms and monomorphies is presumed. In general, this restriction can be lifted too.

Semiotic sequences are equal iff they are decomposable into equal atomic signs, i.e. iff they are atomically equiform and of the same number.

If we take the idea of *decomposability* as the leading strategy for a comparision of sign systems or morphograms we can abstract from the sign repertoires and the singularity of the successor operation. Hence, the test for equality is based on decomposability only.

Compositility and decomposability then can be realized with different operators, e.g. concatenation, chaining, fusion.

Concatenation ($\oplus$): length(A) + length(B) = length(A+B)
Chaining ($\otimes$): length(A) + length(B) = length(A+B)-1
Fusion ($\ominus$): length(A) + length(B) = length(A+B)-n, n >= 2

That is, “*Morphograms are kenomically (morphogrammatically) equal iff they have the same decomposition.”*
Morphograms are kenomically equivalent if they behave (bi)similar under semiotically different interactions.

Semiotic substitution is based on the identification of signs; kenomic transformations are based on the *interaction* between semiotic and monomorphic levels of morphograms.

Prerequisits:
1) Decomposability into parts, based on monomorphies, h,
2) Measurement of the length of sequences, based on iteratie/accretive succession, length(m)∈N,
3) Similarity of monomorphies and morphograms, $m_1 =_{MG} m_2$,
4) Difference between semiotic and kenomic inscriptions, (m) $!=_{sem}$[m],
5) Semiotic disjunctness between substituents and morphogram H, $\text{sem}(m_1, m_2) \cap \text{sem}(H) = \emptyset$.

A monomorphic substitution is correct iff it doesn't violate the the structure (pattern) of the morphogram.
Thus, the substituents of a substitution have to be semiotically disjunct to the morphogram.

Monomorphic substitution

$\forall h,\ m_1, m_2 \in H,\ m_1 \neq_{sem} m_2 :\ Subst_{h/m_1}(H_1) \simeq Subst_{h/m_2}(H_2) \Longleftrightarrow m_1 \simeq_{MG} m_2$

$H_1 = [aabbacc],\ H_2 = [aaccabb],\ H_1 =_{MG} H_2$

$Dec([aabbacc]) = ([aa], [bb], [a], [cc]),$

$h = [aa],\ m_1 = [ddd],\ m_2 = [eee],\ length(m_1) = length(m_2),$

$m_1 \neq_{sem} m_2,\ h \neq_{sem} m_1, m_2,$

$sem(m_1, m_2) \bigcap sem(H) = \emptyset$

$Dec(H_1) = ([aa], [bb], [a], [cc])$

$\longrightarrow Subst(H_1)_{[aa]/[ddd]}([aa], [bb], [a], [cc]) \longrightarrow ([ddd], [bb], [a], [cc])$

$Dec(H_2) = ([aa], [cc], [a], [bb])$

$\longrightarrow Subst(H_2)_{[aa]/[eee]}([aa], [bb], [a], [cc]) \longrightarrow ([eee], [bb], [a], [cc])$

$Subst: ([aa], [bb], [a], [cc]) \longrightarrow ([ddd], [bb], [a], [cc]) =_{MG} ([eee], [bb], [a], [cc])$

$\Longleftrightarrow [ddd] \simeq [eee].$

$\mathbf{([dddbbacc]) =_{MG} ([eeebbacc].}$

Standard representation

$\mathbf{([aaabbcdd]) =_{MG} ([aaabbcdd].}$

Case one :

$h = [aa],\ m_1 = [aaa],\ m_2 = [eee],\ length(m_1) = length(m_2),\ m_1 =_{MG} m_2,$

$sem(m_1, m_2) \bigcap sem(H) \neq \emptyset.$

$sem(m_1 = \{a\},\ m_2 = \{e\}) \bigcap sem(H = \{a, b, c\}) = \{a\},\ (\neq \emptyset).$

$Subst: ([aa], [bb], [a], [cc]) \longrightarrow ([aaa], [bb], [a], [cc]) \neq_{MG} ([eee], [bb], [a], [cc]).$

$\longrightarrow ([aaabbacc]) \neq_{MG} ([eeebbacc])$

Standard representation

$\mathbf{([aaabbacc]) \neq_{MG} ([aaabbcdd])}$

Case two :

$h = [aa],\ m_1 = [ccc],\ m_2 = [eee],\ \text{length}(m_1) = \text{length}(m_2),\ m_1 \neq_{sem} m_2$

$\text{Subst}: ([aa], [bb], [a], [cc]) \longrightarrow ([ccc], [bb], [a], [cc]) \neq_{MG} ([eee], [bb], [a], [cc])$

$\longrightarrow ([cccbbacc]) \neq_{MG} ([eeebbacc])$

$([eeebbacc]) =_{MG} ([aaabbcdd])$
$([cccbbacc]) =_{MG} ([aaabbcaa])$

Standard representation :
$([\mathbf{aaabbcdd}]) \neq_{MG} ([\mathbf{aaabbcaa}])$.

After this descriptive case study, an implementation into a ML program would be a next step to clarify the mechanisms.

Semiotic environment of morphograms

Morphogrammatic transformations can be studied on two levels:
1.On a morphogrammatic level only, say as reflections of produced morphograms,
2. As interactions between semiotics and kenogrammatics.

The range of substitution is defined by the set of marks of the semiotic system involved into the interactions with morphograms.
$sem(m) \in \Omega(\alpha),\ \alpha = \{\alpha_1, \alpha_2, \ldots \alpha_{n-1}, \ \alpha_n\}$.

Hence the range of of kenomic substitution is given by the sign repertoire (alphabet) α and its set of possible concatenations.
For n=2, Ω(α)= {a, b, aa, ab, ba, bb, aaa, ...}.

Direct monomorphic transformations

What happens if the substituents m_1, m_2 are not only semiotically different but also morphogramatically? Is there a reasonable form of substitution possible on the base of different monomorphies? It seems that the equivalence between morphograms under substitution with different substituents is violated.

The condition up to now was that the morphograms are composed by a generalized form of concatenation as iterative and accretive disremption. Hence, the morphogrammatic equivalence between two morphograms supposed equal length of the morphograms. If this condition can be abandoned, a new form of equivalence and substitution could be introduced.

A further abstraction to build equivalences can not refere to the set of signs or kenoms. The only possibility to furthr abstraction has to consider the operations involved. As long as there is only one operation (concatenation) possible an abstraction on it wouldn't make any sense.

Again, *"Morphograms are kenomically (morphogrammatically) equal iff they have the same decomposition"*.

There is no need that the only compositional and decompositional operators are concatenation and decomposition. In fact, the (de)compositional operations in morphogrammatics are including, additional to concatenation, the operation of *chaining* (Verkettung) and *fusion* (Verschmelzung).

$\text{Dec}(H_1) = ([aa], [bb], [a], [cc])$

$\longrightarrow \text{Subst}(H_1)_{[aa]/[ddd]}([aa], [bb], [a], [cc]) \longrightarrow ([ddd], [bb], [a], [cc])$

$\text{Dec}(H_2) = ([aa], [cc], [a], [bb])$

$\longrightarrow \text{Subst}(H_2)_{[aa]/[eee]}([aa], [bb], [a], [cc]) \longrightarrow ([eee], [bb], [a], [cc])$

Given $H_1 = [abba]$ and $H_2 = [aba]$.

How could H_1 and H_2 be morphogrammatically (kenogrammatically) equivalent?

$\text{Dec}([abba]) = ([a], [bb], [a])$

$\text{Dec}([aba]) = ([a], [b], [a])$

That is, how could the two monomorphies $[b]$ and $[bb]$ of H_1 and H_2 be morphogrammatically equivalent?

$\oplus$–Concatenation :

$\oplus : ([ab], [ab]) \longrightarrow \{[abba], [abab], [abbc], [abcd]\}$

$\ominus$–Fusion :

$\ominus : ([ab], [ab]) \longrightarrow \{[aba], [abc]\}$

Hence there are *combinations*, $\oplus$, $\ominus$, such that

$\ominus([ab], [ab]) \longrightarrow [aba], (= A)$ and

$\oplus: ([ab], [ab]) \longrightarrow [abba], (= B)$ with

$\text{length}(A) \neq \text{length}(B)$.

Hence there are *de-fusions*, $\overline{\ominus}$, $\overline{\oplus}$, such that

$\overline{\ominus} : ([aba]) \longrightarrow ([ab], [ba]), (= C_1)$

$\overline{\oplus} : ([abba]) \longrightarrow ([ab], [ba]), (= C_2)$ with

$\text{length}(\overline{\ominus}(A)) = \text{length}(\overline{\oplus}(B)), (C_1 = C_2)$

$\text{Dec}(\overline{\ominus}(A)) = \text{Dec}(\overline{\oplus}(B)) =$

$\text{Dec}([ab], [ba]) = (\text{Dec}([ab]), \text{Dec}([ba]) = ([a], [b], [b], [a]) = ([a], [bb], [a])$

$\text{Dec}(\ominus([ab], [ab])) =$

$\text{Dec}([aba]) = ([a], \text{Dec}(ab]) = ([a], [b], [a])$

$\text{Dec}(\oplus([ab], [ab])) =$

$(\text{Dec}([ab]), \text{Dec}([ba])) = ([a], [b], [b], [a]) = ([a], [bb], [a])$

$\Longrightarrow ([a], [b], [a]) =_{KG/(\oplus,\ominus)} ([a], [bb], [a]) . \Longrightarrow . [b] =_{KG/(\oplus,\ominus)} [bb]$

As a result of the kenomic transformation with the combinations *fusions* and *de-fusions* we get the general formula for equality of morphograms of different length:

$$A =_{MG} B \text{ iff } \bar{\ominus}(A) =_{(\oplus\ominus)} \bar{\oplus}(B).$$

$$Dec\left(\overline{\oplus}(A)\right) =_{(\oplus\ominus)} Dec\left(\overline{\ominus}(B)\right)$$

Concatenation

$$A =_{MG\ conc} B \text{ if } \overline{\oplus}(A) =_{\oplus} \overline{\oplus}(B) \wedge$$
$$length(A \oplus B) = length(B) + length(B)$$

Chaining

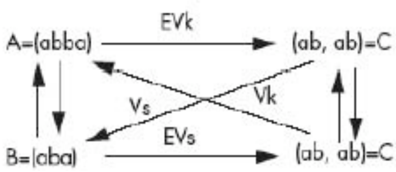
$$A =_{MG\ chain} B \text{ if } \overline{\otimes}(A) =_{\otimes} \overline{\otimes}(B) \wedge$$
$$length(A \otimes B) = length(A) + length(B) - 1$$

Fusion

$$A =_{MG\ fus} B \text{ if } \overline{\ominus}(A) =_{\ominus} \overline{\ominus}(B) \wedge$$
$$length(A \ominus B) = length(A) + length(B) - n$$

Diagrammatic sketch

The following diagram [6] might summarize the idea, again. It goes back to 24.5.1994 when I first sketched the idea and construction. (Ver-Operations: Verkettung, Verknüpfung, Verschmelzung)



Also length([abba]) > length([aba]), mg-equivalent([abba], [aba]) iff EVk * Vs = EVs * Vk.

"*This morphogrammatic equivalence can be compared with the* co-algebraic *concept of bisimulation. Two morphograms are equivalent iff they* behave *the same. This observation maybe the most radical departure from a semiotic understanding of writing."* (Kaehr, From Ruby to Rudy, 2006 , p. 22)

Again, this situation of morphogrammatic behavior gives a hint to an understanding of the fact that Chinese characters are not re-presenting pre-given concepts but are evocating actions. (Kaehr, How to Compose, p. 75, 2007)

What is the difference to semiotic abstractions? The kenomic abstraction happens over the *operators* O- and ⊕ and not over the sign *sets* like for semiotic systems. Equivalence classes in semiotic systems are build over sets of signs and not over the operations on signs. Hence, the kenomic abstraction is a kind of a second-order abstraction. Further studies are included in the paper "Categories and Contextures"[7].

Diamond categorical modeling

A first step towards a categorical construction might be sketched.

$$\left.\begin{array}{ccc}(8)_1 & \longrightarrow & (5+3)_1\\ \Updownarrow & x & \Updownarrow \\ (8)_2 & \longleftarrow & (5+3)_2\end{array}\right\} \; (8)_1 =_{Arith} (8)_2 .$$

Arithmetically the relations $(5+3)_1 \longrightarrow (8)_1$ and $(5+3)_2 \longrightarrow (8)_2$ are well obvious because their relata are all belonging to the same arithmetical systems A_1 and A_2. The situation is getting slightly more intriguing if the relations are belonging to two different arithmetical systems, A_1 an A_2, with $A_1 \cap A_2 = \emptyset$. Hence the relations between $(5+3)_1 \longrightarrow (8)_2$ and $(5+3)_2 \longrightarrow (8)_1$ are of special interest.

The relations (or morphisms) $(5+3)_1 \longrightarrow (8)_2$ and $(5+3)_2 \longrightarrow (8)_1$ can be seen as *translational* morphisms between two *discontextural* arithmetical systems A_1 an A_2. Hence, a possibility of a *comparison* between $(8)_1$ and $(8)_2$ is established, which is demanding its own third contexture to take place.

This little example is of interest independently of the numeric values used and the definition of their axiomatics.

After all, the construction of an equivalence of morphograms of different length might be set into a more intelligible formalism with the help of *diamond category theory*, thus diamond constructions of categorical sums and products shall be used.

$$\begin{array}{ccc} & \mathbf{X}_2 & \\ \forall_2 \diagup & \exists!_2 \downarrow & \diagdown \forall_2 \\ \end{array}$$
$$\left[\begin{array}{ccccc} A_2 & \longleftarrow & A_2 \oplus B_2 & \longrightarrow & B_2 \\ A_1 & \longrightarrow & A_1 + B_1 & \longleftarrow & B_1 \end{array}\right]$$
$$\begin{array}{ccc} \forall_1 \diagdown & \exists!_1 \downarrow & \diagup \forall_1 \\ & \mathbf{X}_1 & \end{array}$$

The aim of the diamond category construction is to construct and compare X_1 and X_2. The *category* part is covering the construction of X_1, with $\ominus : A_1 + B_1 \longrightarrow X_1$, the *saltatory* part is covering the construction of X_2, with $X_2 \longrightarrow B_2 + B_2$. Both parts of the diagram are complementary and commutative.

The interaction between categorial and saltatorial parts of the construction might be set into a chiastic interplay.

$$\text{Chiasm}\left(\left(A, B\right), \left(X_1, X_2\right), \oplus, \ominus\right):$$
$$\left(\begin{array}{cccc} \ominus : & (A, B) & \longrightarrow & X_1 \\ & \updownarrow & x & \updownarrow \\ \oplus : & X_2 & \longrightarrow & (A, B) \end{array}\right)$$

What's the fuss for?

As a radical result for a new conception of Web *mobility* we get the possibility of a new type of mobility in the static addresses of UANs, URLs and IPs even independently of any physical or informatical movements of the actors. That is, the *statics* of common Web mobility concepts with their hierarchic structures have to be *dynamized* to offer a free Web mobility in a Knowledge Grid. But this is asking for a high price: the sacrifice are our natural number systems which are guaranteeing *universal* and *unique* addressing methods. In fact, its only the proclaimed *hegemony* and *uniqueness*, and not the number systems as such, which have to be transformed and disseminated. Without saying, the whole apparatus of classical, i.e. informatical Web mobility concepts are saved and are getting their placement in the new paradigm of a kenogrammatically designed organizational structurations of mobile knowledge grids.

2. Web mobilities

2.1. Mobile computing

2.1.1. Many Faces of Mobility

Transport, translation, chiasm, worldmodels, kenomic transitions, metamorphosis.
Semiotic spaces (Goguen).
Kenomic spaces (Skizze-0.9.5).

2.1.2. The Actor Model

Agha's new model[8] is introducing a highly complex strict hierarchy of URLs with the assistance of meta-actors helping the brave basic-actors, based on supressed primitive-actors[9], of the Actor system to behave communicatively in a mobile informatical environment.

"A naming service is in charge of providing object name uniqueness*,* allocation*,* resolution*, and* location *transparency. Uniqueness is a critical condition for names so that objects can be uniquely found given their name. This is often accomplished using a name context. Object names should be object location-independent, so that objects can move preserving their name. A global naming context supports a universal naming space, in which context-free names are still unique. The implementation of a naming service can be centralized or distributed; distributed implementations are more fault-tolerant but create additional overhead."* Gul A. Agha, Carlos A. Varela, Worldwide Computing Middleware

The architecture of global naming is given in extenso by Agha[10].

"*Worldwide computing systems require a scalable and global naming mechnism. Moreover, the naming mechanism must facilitate object mobility; this implies that the object name should completely abstract over the location of an object, so the migration does not break existing references. Contrast this to the Web infrastructure, which users location-dependent references (UROLs) therby inhibiting transparent document relocation."* (Varela)

This naming abstraction is in direct opposition to the kenomic abstraction of the identity/locality relation.
To *"completely abstract over the location of an object"* is eliminating the inter-relationship between identity and locality of an object, which is basic to kenomic mobility.
.
Abstraction as call-by-name, is naming. Naming is identifying an object. The process of naming happens in a context which is not part of the abstraction. Naming is a special kind of abstraction as identification, hence called is-abstraction. The is-abstraction is the fundamental abstraction of the lambda calculus.

A general concept of abstraction is thematization. Thematization is evocating an object without identifying it by naming. Hence the object shall be called phenomen. Thematization is enabling complex and mediated actions of naming, depending on different view-points and reflecting contexts of the phenomen be named. Such a kind of abstraction is called as-abstraction. Obviously, the interplay between different standpoint-depending naming actions is not itself a naming action but *thematization*.

Space and Place for Actors and Agents
"If the locatedness for a classic actor in his midleware theater is an URL, based on URI, etc., thus a fixed identity address, then the locatedness of a contextural agent is the morphogram of such an address. The morphogram of the locatedness of an Agent is guaranteeing the liveliness *of the Agent and is preventing it to be considered as a physical object. An Agent is a reflectional/interactional unit and therefore not addressable and nameable by a single and simple identity producing and identifiable name. An Agent can have a name but it isn't a name.*
Classic Actors are much more defined by their name and their name is used as if it would be the Actor. In this sense an Actor is a name and is not just having a name. An Actor is defined by a name-giving abstraction, i.e., the is-abstraction." (Kaehr, Actors+Objects, 2007)

2.1.3. The Bigraph Model

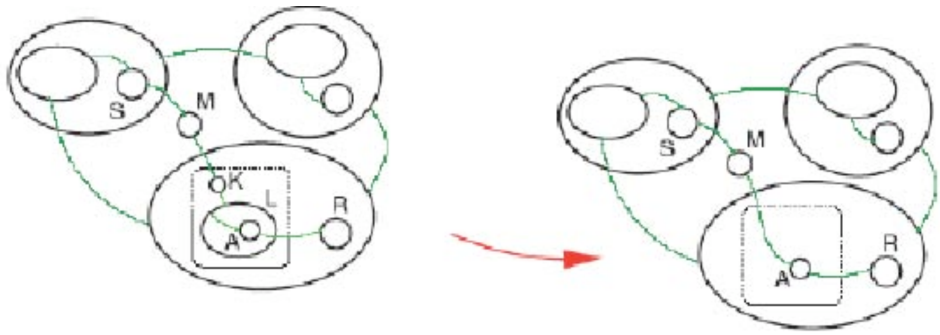
Locality and connectivity in a communicational space are designed by Milner's bigraph model.

"Our strategy here is to tackle just two aspects of mobile systems simultaneously: mobile *locality* and mobile *connectivity*. Already this combination presents a challenge: to what extent are locality and connectivity interdependent? In plain words, does where you are affect whom you can talk to? The answer must lie in the level of
modelling. To a user of the Internet (seeing it abstractly) there is total independence, and we want to model it at a high (i.e. abstract) level, just as it appears to users. But to the engineer these remote communications are not atomic; they involve chains of interactions between neighbouring entities, and we must also provide a low-level model which reflects this reality. These two levels must surely be part of a single multi-level model that explains how higher levels are realised by lower levels."

"Bigraphical reactive systems are a model of information flow in which both *locality* and *connectivity* are prominent. In the graphical presentation these are seen directly; in the mathematical presentation they are the subject of a theory that uses a modest amount of algebra and category theory. A bigraph may reconfigure both its locality and its connectivity. The example pictured above shows how reconfiguration is defined by reaction rules; in that case, the rule may be pictured thus:

Key metaphorics in the bigraph agent model is the *key* with its locking and unlocking functionality.

"The [next] picture illustrates how physical and virtual space are mixed. It represents how a message **M** might move one step closer to its destination. The three largest nodes may represent countries, or buildings, or software agents. In each case the sender **S** of the message is in one, and the receiver **R** in another. The message is en route; the link from M back to **S** indicates that the messages carries the sender's address. **M** handles a key **K** that unlocks a lock **L**, reaching an agent **A** that will forward the message to **R**; this unlocking is represented by a reaction rule that will reconfigure the pattern in the dashed box as shown, whenever and wherever this patterns arises."



Milner, Robin (2005): BIGRAPHS: A TUTORIAL, April 2005, Beijing[11]

2.1.4. A Transitional Model

There is a transitional [12] approach to mobility too. It takes a higly speculative stance to promote a transition from the *informatical* to the *knowledge* paradigm of mobile computing in polycontextural worlds. Both, the Actor Model and the Bigraph Model, are founded, more or less, in category theory and its underlying semiotics. The transitional model tries to surpass the conceptual and formal limits imposed by category theory and semiotics with the help of the emerging *diamond strategies*.[13]

"From a model of interactions to a design of interactionality, the transitions to be risked might be:
From the global, ubiquitous and universal Web of computation to the kenomic grid *of pluriversal contexturality containing the chiasm of global/local scenarios.*

From the locality in the Actor model of informatical events to the positionality *of contextures in the kenomic grid, positioning informatic localities.*

From the mobility in the Actor model of informatical flows between ambients (context, locality) of the same contextural (ontological, logical, semiotic) structure to a metamorphosis *between contextures, augmenting complexity/complication of contextural scenarios implementing clusters of informatical ambients and mobility.*

From the operations between actional ambients to the operationality *in polycontextural situations realized by the super-operators (identity, replication, permutation, reduction, bifurcation) placing ambient operations into the grid.*

From the connectivity of actions at a locality of message-passing, using a key to unlock a lock of an agent, to different kinds mediation *between contextures containing informatical connectivity.*

These transitions seems to record a catalogue of minimal *conditions to be fulfilled to realize interactionality/reflectionality and interventionality in such complex constellations as the emerging knowledge grid."*

2.1.5. The key of mobile computing

Global URL and the identity of the key. The key of naming and the naming of the key.
The common semiotic presupposition of mobile computing is grounded in graphemic (or syntactic)
identity. Such a sentence sounds trivial because it is not only well known but the *sine qua non* of any scientific formulation. But its triviality is dissolved if it is localized into the graphemic chain of epochs from the pre-semiotic to

the semiotic and further to the post-semiotic epoch as a trans-semiotic statement.

Mobility in identity systems is restricted to such identity; identity is guaranteeing security and global control for the prize of structurally restricted mobility. That is, mobility in identity systems happens under the roof of the identity of the global URL, e.g. Agha's UAN. Hence, despite its security it is the most vulnerable guarantee. A successful attack on its identity turns the system down.

2.2. Kenomic mobility

URLs, obviously, are words, representing numerical words. Numeric words are frozen kenomic scriptures.

Grammatologically speaking, the hierarchy between spoken and written language is inverted in graphematic systems. Hence, scriptures are not conceived as the inscriptions of words seen as connected with the thought, soul and life of a subject, they are therefore soul-less, dead, erratic (Platon, Roussau, de Saussure) but much more the life of spoken words is the witness of death, words are caged into the coffin of identity.
In other words, kenomic inscriptions are beyond life and death; spoken words are the words of death.

Mobility is a minimal condition for life.

The singularity and individuality of a kenomic address is given by its history. This is not a pre-given act of decision by an administrational priority but a mainly unknown and hidden determination of the address developed in the process of addressing.

Ideally, addresses are emerging by addressing. Hence, there is no key without using the key. Keys are not pre-given and representing a codification but are co-created during the process of addressing, co-creating the address of the key's addressing.

How many keys?

The identification number sequence given to a customer of a kenomic system is not a numeric number anymore but the dynamic pattern of of a number sequence, i.e. a *morphogram*. Hence a multitude of different concrete numeric addresses are given to the addresser. The system is not dealing with those addresses in concreto but with their pattern alone. That is, the URL is computed morphogrammatically and not numerically by the administrational instance.

The metaphor is: many keys to un/lock one lock.
Is this augmenting or reducing security? Mobility? Dynamics?
If a lock can be opened by many keys it is surely easier to crack the code and open the door than with a single and unique key. Hence polysemic locks are easier to crack than monosemic locks.
But this is missing the argument! No single key is unlocking the lock. Only the underlying morphogram of the different keys is unlocking the lock. Thus, the question is, how to get access to the morphogrammatic lock? Is it accessible at all? There is no direct path from the single numeric keys to the unique morphogrammatic lock. Uniqueness in morphogrammatics is not connected with identity, like for numeric keys. Keys and locks in classical systems are both part of identity systems; they share the same semiotic abstraction.

The paradox is: The more keys are at hand the more difficult it is to un/lock the lock.
And at once, obviously, the more locks to be un/locked by one key the more difficult it is to find the key.

A key is not only an opener but with its cryptographic possibilities a discloser.
The more keys the more complex the cryptography. The complexity is not only in the numeric keys but in their multiple decompositions. There is a double or second-order cryptography involved.

The more keys fits the lock the less complex is the lock. The more keys *possible* to unlock the lock the less complex the lock.
The more keys *necessary* to un/lock the lock the more complex is the lock.

The more keys that don't fit the lock the easier it is to unlock the lock.
The more keys there are which don't fit the lock the easier it is to unlock the lock.
The more keys there are which fit the lock the harder it is to unlock the lock.
The more keys needed to unlock the lock the more complex the lock.

The key to mobile keys

The key to mobility seems to be mobile keys. Locality and connectivity in the sense of Web mobility are of second interest in keomic systems. Kenomic mobility is possible and reasonable even without any factual physical or informatical mobility by an actor. A static actor system should be able to be involved into the dynamics of mobility of the knowledge grid without getting forced to physical and informatical mobility.

The order of statics and dynamics of Web mobility is reversed in kenomic systems . Today, actors are mobile and the organizational institution is immobile and guaranteeing physical and informatical addressability and mobility. The concept or paradigm I'm halluzinating for is dynamic mobility and metamorphic transformability of the, until now, static organizational system of mobile computing.

Dynamic keys are offering mobility even for static actors. This doesn't sound absurd if we connect kenomic mobility with addressability and security. If the key is mobile, i.e.dynamic in a kenomic way, an attack to crack the code of the key turns into absurdity. That is, the abstraction from the locality of actors, like in Aga's model, can be understood in an *inverse* manner. *"Location-independence"* in Agha's model is connected with the mobility of an actor in the physical and informatical world, in a kenomic sense, *"location-independence"* has a rejectinal meaning: independence from the *necessity* of mobility between locations. And on the other hand, acceptance of "location-independence" for the constitutive difference of existence (occurrence) and locality (positionality) of events.

A location-independent system has two main features: a) it is *blind* to the fact that it is itself located, b) it is *blind* to the fact that it necessarily doesn't have an environment. Hence it is helpless against any attack or positive surprise from the *otherness* of itself.

Because of their dynamics, dynamic keys are not universal and unique, they don't have *"worldwide uniqueness"*, their world is not uni-versal but *pluri-versal*, their uniqueness is not identifiable by universal naming. Kenomic keys are situational, historical and depending on contextual use, learning reflectionally and interactionally to change and redefined self-determination.

"Since universal actors are mobile--their location can change arbitrarily--it is critical to provide a universal *naming system that guarantees that references remain consistent upon migration."*

"A Universal Actor Names (UAN) refers to an actor during its life-time in a location-independent *manner. The main requirements on universal actor names are location-independence, worldwide uniqueness, human readability, and scalability. We use the Internet's Domain Name System (DNS) [Mockapetris, 1987] to hierarchically guarantee name uniqueness over the Internet in a scalable manner."* (Agha, Varela)

From an *epistemological* point of view, I still have to insist on the crucial difference of *surface*- and *deep*-structure. Informatical theories and methods to deal with Web mobility are dealing with the surface-structure of Web activities. The kenomic approach tries to reflect and interact with the statics of the deep-structure of the Web. Both, surface- and deep-structure together have to be addressed simultaneously to develop a paradigm of an evolving knowledge grid.

The whole exercise experimented in this paper is trying to deconstruct the presuppositions of Mobile Computing: *uniqueness*, *universality*, *identity*, *human-readability,* etc. Such a manoeuvre might uncover some hints for a new paradigm of mobilities (plural!) in a pluri-versal knowledge grid.

3. Architectonics of kenomic Mobility

3.1. Architectonics of kenomic Actor systems

1. *Primitive actors* are zero-order actors, they are not allowed to interact but are responsible for the whole actor system to work properly, i.e. without paradoxes and circularity.
    a. Primitive actors are not active on the stage or arena but at the back-stage. Primitive actors are hidden actors.
        i. Primitive actors are enabling the interactional actions of basic actors. Without the support by primitive actors self-destructive actions of infinite regress, antinomic cricularities (paradoxes) are unavoidable in classical, i.e. monocontextural actor systems.
        ii. Primitive actors are typical for monocontextural (formal) systems.
2. *Basic actors* are first-order actors, their definition is to interact with other actors of an actor system.

a. Basic actors are the actors on stage. They are playing the big interactional drama on a single arena.
    i. Basic actors are playing on stage on the base of the hidden support by primitive actors.
    ii. Basic actors are playing on stage on the prospect of the open guidelines by meta-actors.

3. *Meta-actors* are second-order actors, they are responsible for the interactivity between different actor systems in a global actor system, like the WWC (World Wide Computing).
    a. Meta-actors are the directors of the actor play. They manage the interactions between the actors, the actor systems and their universal distribution in a global interactional game. Hence, on a higher level they are also the organizational committee of the distributed actor systems.
    b. This reflectional capacity of the meta-levels of second order systems can be iterated to meta-levels of the second-order system. That is, in the second-order systems, meta-reflections (introspection) can be iterated without changing the second-order status of the system. No meta-reflection leads to a third-order system. No iteration of meta-reflection has to collapse into first-order systems.
        i. Meta-Actor systems, which are not yet embedded into the Diamond Actor system are not immun against the infinite regress problem imposed by the infinite iterability of meta-reflections.
    c. Deepness of meta-reflections of second-order systems vs. broadness of object-reflection of first-order systems.
    This defines the reflectional Actor system for uni-versal interactions as it is exposed by Agha’s middleware approach.
4. *Trans-actors* are third-order actors, they are disseminating second-order actor systems over the kenomic matrix of polycontextural interactions. Polycontextural interactivity is pluri-versal.
    a. trans -actors in polycontextural systems are represented by the so-called super-operators (identity, permutation, reduction, replication, bifurcation) defining operationally the interactionality between disseminated universal actor systems.
    b. trans-actors are the *mediators* between disseminated actor systems. Mediators are the organizers of the interplay of different primordial actor systems.
        i. Interactivity between disseminated actor systems is ruled by the mechanism of chiasms.
        ii. Chiasms are combining order-, exchange- and coincidence-relations between actors and actands on different levels of polycontexturality.
        iii. As a consequence of the chiastic structure of disseminated actor systems the primitivity of the primitive actors is resolved into a contextural relativity. What functions as a primitive in one contexture functions as a non-primitive in a neighbor contexture, and vice versa.
        iv. Hence, problems of circularity are *restored* at the situation of any single elementary contexture and *resolved* by the distribution of the construction of chiastic circularity over different contextures.
5. *Diamond-actors* are forth-order actors, they are embedding the activities of the trans-actors into diamonds.
    a. Diamond-actors are enabling complex disseminated actor systems to incorporate the possibility of the new as the otherness of the actor system.
    b. Diamond actors are playing a double role. They are responsible for the mobility system and are enabling its environment. The environment of a mobility system is the place of the otherness. This can incorporate attacking events and/or the surprise of the new.
6. Diamond actor systems are localized and positioned into the *kenomic matrix*.

a. The kenomic matrix is opening up spaces to general actor systems to place interactional, reflectional and interventional activities.

3.2. From hierarchy to a heterarchy of diamond actor systems

The classic hierarchy of the tectonics of actor systems is given by the hierarchy of:

$$AS = \left[\text{meta} \left[\text{basic} \left[\text{primitive} \left[\text{actors} \right] \right] \right] \right].$$

The original Actor Model is based on actors only. *"Everything in an Actor Model is an actor."* (Hewitt)

As it is well known, this everything-is-ism leads quickly to unpleasant consequences, which can hardly be accepted, especially from a computer science point of view. The unpleasant species are '*illustre'* guests of many departments, they are called "vicious circles", "infinite regress", "paradoxes", "antinomies" and they got even a trendy appearance as "circulus creativus".

Hence, something has to be done. For that the brave "primitive actors" got a role in the play. Sometimes they experience the privilege of being tolerated, domesticated and baptized as "base actors" of a special kind.

On the base of existing presumptions of rationality and its mono-contextural constitution there is no escape in sight to such a situation. We can reject or forbid antinomies or we can try to domesticate them into a save corner of the hierarchical kingdom of reasoning and computation.

That's obviously very boring!

Therefore, I'm opting for a polycontextural and diamondal undertaking, adventurous or not.
The exercise to risk is quite simple:
Transform any circle or circularity into chiasms, first, then complete the chiasms towards diamonds of polycontextural frameworks!

3.2.1. Actor systems as mono-contextural reductions

$$AS_{mono} = \begin{bmatrix} \text{Meta} \\ \begin{bmatrix} \text{Basic} \\ [\text{Primitive}] \end{bmatrix} \end{bmatrix}.$$

This kind of a hierarchy can be seen as a reduction from the polycontextural diamond actor system to a monocontextural categorical actor system. That is, for matrix=1, diamond=0 and transoperations=0, the diamond Actor System is reduced the the monocontextural actor system AS_{mono} with its singular hierarchic distinction of meta/basic/primitive actors.

$$\begin{bmatrix} \text{Matrix} = 1 \\ \begin{bmatrix} \text{Diamond} = 0 \\ \begin{bmatrix} \text{Trans} = 0 \\ \begin{bmatrix} \text{Meta} = 1 \\ \begin{bmatrix} \text{Basic} \\ \left[\text{Primitive} \right] \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{Meta} \\ \begin{bmatrix} \text{Basic} \\ \left[\text{Primitive} \right] \end{bmatrix} \end{bmatrix}.$$

Hierarchic solution of circularity
"The actor model is completely *uniform*. It includes a single kind of entity, actors, just as the Smalltalk-80 model only includes objects. [...]

"This uniformity raises the problem of *infinite regress*: if any access to information should be performed by message sending, messengers themselves would have to send a message in order to access to the message they carry and would deliver it to the receiver, and so forth.
So-called *primitive* actors, which do not need to send a message to respond a request, are provided to deal with this difficulty." Massing et al., Object-Oriented Languages, 1991, p. 299

"To avoid infinite regress of delegation, so-called *rock bottom* actors never delegate and do not send messages. They correspond to the *primitive* actors of the Hewlett's model and represent entities of a few specific types: for example, numbers, symbols and lists, in the lisp implementation. Their script is held by the interpreter." (ibid., p. 312)

Explanation of the brackets

1. **Matrix=1**: AS_{mono} is taking place in a scriptural space but their is no need to be aware of it and to notify it. To defend a position is possible or necessary only if their are more than one position involved. Hence, mono-contextural systems are blind of possible neighbor systems and of the fact of being positioned, i.e. occupying a locus in a scriptural design.
2. **Diamond=0**: There is no need to involve AS_{mono} into a complementary interplay between categories and saltatories. Categories with their unsplit mono-contexturality are enough. Everything else is disturbance, creating fear.
3. **Trans=0**: Trans-operators are guiding the metamorphic interactions between contextures in a polycontextural complexion. Hence, in a mono-contextural situation, trans-operators are reduced to the identity operation, which again, can be omitted because it simply states the self-identity of the system with itself.
4. **Meta=1**: Finally, the Actor model for mobile computing gets its director. One, obviously, is enough to rule the (hidden) hierarchy of base and primitive actors.
5. **Basic=1:** The Actor model with its director rules the uniquenes of a singular base actor system which is secured by a unique troup of primitive actors.
6. **Primitive=1:** Primitive actors are building a unique system of core actors, preventing possible troubles, such as circularity produced by the base actors, .

3.2.2. Chiastic AS: basic/primitive

The hierarchy of primitive and basic actors, necessary in mono-contextural systems to avoid circularity, is transformed into a chiasm between the functionality of actors as *primitive* and as *basic* actors.

$$\left[\begin{array}{l}\text{Matrix}=2\\ \left[\begin{array}{l}\text{Diamond}=1\\ \left[\begin{array}{l}\text{Trans}=1\\ \left[\begin{array}{l}\text{Meta}=1\\ \left[\begin{array}{l}\text{Basic}\\ \left[\text{Primitive}\right]\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right] \Longrightarrow \left[\begin{array}{l}\text{Matrix}\\ \left[\begin{array}{l}\text{Diamond}\\ \left[\begin{array}{l}\text{Trans}\\ \left[\begin{array}{l}\text{Meta}=2\\ \left[\chi\left(\text{Basic}/\text{Primitive}\right)\right]\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right] =$$

$$\left[\begin{array}{l}\text{Matrix}\\ \left[\begin{array}{l}\text{Diamond}\\ \left[\begin{array}{l}\text{Trans}\\ \left[\begin{array}{l}\text{Meta}^1\\ \left[\left(\text{Basic}\longrightarrow\text{Primitive}\right)\right]\end{array}\right]\updownarrow\left[\begin{array}{l}\text{Meta}^2\\ \left(\text{Basic}\longrightarrow\text{Primitive}\right)\end{array}\right]\end{array}\right]\end{array}\right]\end{array}\right]$$

$$\text{Chiasm}^{(2,1)}=\chi\left[\text{Meta}^1,\ \text{Meta}^2,\ \text{Basic},\ \text{Primitive}\right]$$

$$\left[\begin{array}{l}\text{Chiasm}^{(2,1)}:\\ \quad\text{Meta}^1:\left[\text{Basic}\longrightarrow\text{Primitive}\right]\\ \qquad\qquad\updownarrow\quad x\quad\updownarrow\\ \quad\text{Meta}^2:\left[\text{Primitive}\longleftarrow\text{Basic}\right]\end{array}\right]$$

A chiastic solution of the *"infinite regress"* problem is possible only for the cost of the commodity of the mono-contextural design, which has to be sacrificed to the dynamics of polycontextural systems.

Hence the chief director of the meta-operators has to be split into a cooperation of two directors building together the directors *team* of the cooperating theaters. This sacrifice of power opens up space to distribute the vicious circularity of self-referential base actors over two loci to form out of the circle a chiasm between base and primitive actors and the positions meta1 and meta2. Such a sacrifice shouldn't be too hard: their are still the directors dominating the distributed and mediated base and primitive actors. No director has to appear on stage with the actors. But also, no primitive actor has to stay hidden behind stage. The new fun of the game is the inter-exchange between back-stage and on-stage appearances of actors on the arena of different theaters.

3.2.3. Chiastic AS: trans/meta/basic/primitive

$$\begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}\\ \begin{bmatrix}\text{Trans}\\ \begin{bmatrix}\text{Meta}\\ \left[\chi\left(\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\end{bmatrix}\end{bmatrix}\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}\\ \begin{bmatrix}\text{Trans}\\ \left[\chi\left(\text{Meta}/\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\end{bmatrix}\end{bmatrix} =$$

$$\begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}\\ \left[\begin{bmatrix}\text{Trans}^1\\ \left[\left[\text{Meta} \longrightarrow \left(\text{Basic}/\text{Primitive}\right)\right]\right]\end{bmatrix}\right] \updownarrow \left[\begin{bmatrix}\text{Trans}^2\\ \left[\text{Meta} \longrightarrow \left(\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\right]\end{bmatrix}\end{bmatrix}$$

But that’s not enough. Cooperations are still taking part in an old fashioned modern world view. But in-between post-modern experiences have led to a fundamental involvement of the directors into the play they have to direct, now becoming aware, that there was always an interaction between the brave players and the directors. Hence, a new step of the interchangeability interplay is opened: directors are becoming actors and actors are becoming directors.
All that is well operated by the supremacy of the trans-operators. But again, directors have to share their supremacy between each other.

There is some advantage, they learned to do it from the directors and they still can be assured of the common insurance by the diamond actors, who are well positioned in the kenomic matrix.

Because of the success of the play the difference of base and primitive actors gets neglected. This happens to all systems where the slaves are well domesticated.

Meta-actor and base actor
"Interaction between meta-actor and base actor.” (Varela)

"We use the meta-actor[14] extension of actors to provide a mechanism of architectural customization. A system is composed of two kinds of actors: *base* actors and *meta-actors*. Base actors carry out application-level computation, while meta-level actors are part of the runtime system (middleware) that manages system resources and controls the base-actor’s runtime semantics.” (Agha, Varela)

"Thus, in a reflective architecture, a system is composed of two kinds of actors--*base*-level (application) actors and *meta*-level actors or meta-actors.” (ibd. p.12)

$$\begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}\\ \text{Trans}\\ \left[\chi\left(\text{Meta}/\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}\\ \left[\chi\left(\text{Trans}/\text{Meta}/\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\end{bmatrix} =$$

$$\begin{bmatrix}\text{Matrix}\\ \begin{bmatrix}\text{Diamond}^1\left(\text{Category}\right)\\ \left[\text{Trans} \longrightarrow \left(\text{Meta}/\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix} \updownarrow \begin{bmatrix}\text{Diamond}^2\left(\text{Saltatory}\right)\\ \left[\text{Trans} \longrightarrow \left(\text{Meta}/\text{Basic}/\text{Primitive}\right)\right]\end{bmatrix}\end{bmatrix}$$

Until now the distribution was guaranteed by the mechanism of polycontexturality.
No involvement of the intriguing apparatus of diamonds was necessary to involve trans-operators and directors of the meta-actors into the play.

3.2.4. Chiastic interaction AS: diamond/mobile

$$\text{Chiasm}\left(\left[\begin{matrix}\text{Matrix}\\ \left[\begin{matrix}\text{Diamond}\\ [\text{Mobile}]\end{matrix}\right]\end{matrix}\right]\right) \Longrightarrow$$

$$\left[\begin{matrix}\text{Matrix}\\ [\chi(\text{Diamond}/\text{Mobile})]\end{matrix}\right] = \left[\begin{matrix}\text{Matrix}^1\\ [\text{Diamond} \longrightarrow \text{Mobile}]\end{matrix}\right] \Big\|\!\uparrow\!\downarrow\!\Big\| \left[\begin{matrix}\text{Matrix}^2\\ [\text{Diamond} \longrightarrow \text{Mobile}]\end{matrix}\right]$$

$$\text{Mobile} = \left[\begin{matrix}\text{Meta}\\ \left[\begin{matrix}\text{Basic}\\ [\text{Primitive}]\end{matrix}\right]\end{matrix}\right], \ \text{Trans}: \text{Mobile} \longrightarrow \text{Mobile}$$

A chiasm between *mobile* and *diamond* seems not easy to grasp.

The diamond rules the positioning of *mobile* in the kenomic matrix. The mobile, antidromically, is enabling the structure of such positioning of the mobile by the diamond. The process-structure, i.e. the structuration of mobile and diamond are inter-related, inter-woven and building together the chiastic interactional Actor system.

A further simple step in the conceptualization of Mobile Actor Systems is introduced by the diamondzation of the chiastic approach to the diamond/mobile interaction.

Additional to the acceptional patterns of the (pure) chiasm, the rejectional behaviors shall be involved. The acceptional patterns are reflecting the dynamics of the pure chiasm into an own domain.This domain (contexture) is representing the "*what*" of the chiasm, while the chiasm itself is inscribing the "*how*" of the interaction. The rejectional behaviors are mirroring the antidromic, enantiomorph, 'inverse' patterns of the chiasm.

Diamond Actor Systems

$$\text{Diam}\left(\left[\begin{matrix}\text{Matrix}\\ \left[\begin{matrix}\text{Diamond}\\ [\text{Mobile}]\end{matrix}\right]\end{matrix}\right]\right) \Longrightarrow$$

$$\left[\begin{matrix}\text{Matrix}\\ [\delta(\text{Diamond}/\text{Mobile})]\end{matrix}\right] = \left[\begin{matrix}\text{Matrix}^1\\ [\text{Diamond} \longrightarrow \text{Mobile}]\end{matrix}\right] \Big\|\!\Updownarrow\!\Big\| \left[\begin{matrix}\text{Matrix}^2\\ [\text{Diamond} \longrightarrow \text{Mobile}]\end{matrix}\right],$$

$$\left[\begin{array}{l}\text{Diam}^{(2,1)}\\ \quad \text{Pos}^1 : [\text{Diamond}] \doteq [\text{Diamond} \longrightarrow \text{Mobile}] \doteq [\text{Mobile}]\\ \qquad\qquad \downarrow \qquad\qquad \updownarrow \quad \times \quad \updownarrow \qquad\qquad \uparrow\\ \quad \text{Pos}^2 : [\text{Mobile}] \doteq [\text{Mobile} \longleftarrow \text{Diamond}] \doteq [\text{Diamond}]\end{array}\right]$$

Nevertheless, the whole drama, as just sketched, is one and only one *thematization* of the possibilities of mobile computing based on diamonds and polycontexturality inscribed into the kenomic matrix.

Mobility

"There are three types of mobility in distributed systems: resource *migration to improve locality of access,* code *migration for dynamic application behavior, and* user *mobility - multiple points of application access."* (Varela, p. 4)

Polarizations

Resources: Abstractness of classical semiotic systems vs. kenomic concreteness of morphogrammatic inscriptions.
Code: Universal code for migration and dynamics vs. pluri-versal thematizations for metamorphosis and interchange.
Users: Ego-based mobility in a homogeneous physical and informatical space vs. interactionality/reflectionality based interplay in a kenomic matrix.

3.2.5. Disseminated ASs

$$\begin{bmatrix} \text{General} - \text{Matrix}^{(m,n)} \\ \begin{bmatrix} \text{Diamond}_{1.1} \\ [\text{Mobile}] \end{bmatrix} \begin{bmatrix} \text{Diamond}_{1.2} \\ [\text{Mobile}] \end{bmatrix} \ldots \begin{bmatrix} \text{Diamond}_{1.n} \\ [\text{Mobile}] \end{bmatrix} \\ \begin{bmatrix} \text{Diamond}_{2.1} \\ [\text{Mobile}] \end{bmatrix} \begin{bmatrix} \text{Diamond}_{2.2} \\ [\text{Mobile}] \end{bmatrix} \ldots \begin{bmatrix} \text{Diamond}_{2.n} \\ [\text{Mobile}] \end{bmatrix} \\ \ldots \quad \ldots \quad \ldots \quad \ldots \\ \begin{bmatrix} \text{Diamond}_{m.1} \\ [\text{Mobile}] \end{bmatrix} \begin{bmatrix} \text{Diamond}_{m.2} \\ [\text{Mobile}] \end{bmatrix} \ldots \begin{bmatrix} \text{Diamond}_{m.n} \\ [\text{Mobile}] \end{bmatrix} \end{bmatrix}$$

Disseminated Actor systems (DAS) are necessary to handle polycontextural constellation of the knowledge grid. Knowledge, hence is conceived as categorial strictly different from concepts and strategies like information, data, informatic objects which are all defined mono-contexturally, i.e. independent from the reflectionality and interactionality of the co-creating participant and designer of a knowledge grid. A grid or a mesh is not a web.
A grid is understood as a mediation of distributed contextures (of meaning) which are arranged, interactionally and reflectionally, into a multi-layered and heterarchic complexions.
Therefore, a single Actor system is not covering the complexity of knowledge but the uniformity of a informatical domain only.

4. Agha's UAMs in the Matrix

4.1. How to map universals onto the matrix?

"The universal actor model [UAM] *extends* the actor model [Agha, 1986] by providing actors with *universal* names, location awareness, remote communication, migration, and limited coordination capabilities [Varela, 2001]." (Agha, Varela, p. 6, 2004)

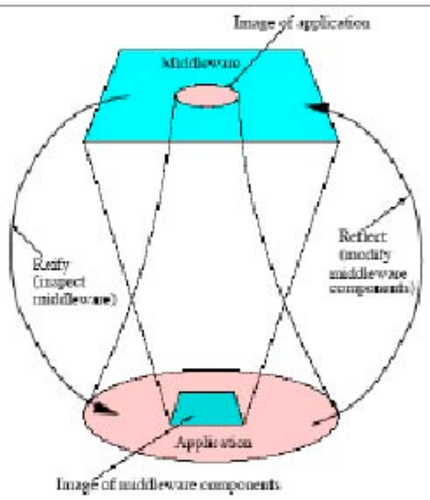


Figure 1.2: Using reflection, an application can inspect and modify middleware components.

Sometimes, a diagram offers more information about the modeling philosophy than the verbal and formal descriptions of it. At least, I don't easily see the hierarchic structure between *application* and middleware [Agha, Varela 2004] as it is proposed in Varela's text. Astonishingly, it looks much more like a Yin-Yang-Figure than a hierarchic diagram.

Infinite regress, again?

It seems not yet been reflected by the UAM approach to reflection that the *infinite regress* problem, or is it a progress?, of reflection and meta-reflection of meta-level actors (meta-actors) occurs automatically in meta-reflectional systems again as it happened for the base actors who are depending on the stopping facilities of the primitive actors. As it was necessary to introduce the regress stopping primitive actors as *rock bottom* actors it seems necessary to introduce a new kind of regress stopping ultra-meta actors as *top rocket* (or *high sky*) actors to stop the system transcending into unknown horizons.

Reflection of Application/Middleware = χ[Reflect, Reify, $Image_{Appl}$, $Image_{Middle}$].

$$\left[\begin{array}{l} \text{Chiasm}^{(2,1)}: \\ \quad \text{Reflect}^1 : \left[\text{Application} \longrightarrow \text{Middleware}\right] \\ \qquad\qquad\qquad \updownarrow \quad \times \quad \updownarrow \\ \quad \text{Reify}^2 : \left[\text{Middleware} \longleftarrow \text{Application}\right] \end{array}\right]$$

The modeling strategy to map the Universal Actor Model (UAM) onto the kenomic matrix is quite simple: Universals are mapped into pluri-versals. That is, the *uniqueness* of the Universal Actor Model has to be disseminated over the pluri-versal matrix of polycontexturality. As a consequence of such a distribution and mediation new interactional and reflectional mappings between the distributed UAMs are introduced.

As a result, some modeling of the original UAM might be reframed into other strategies. For example, the modeling of reflection as *"Using reflection, an application can inspect and modify middleware components."* might get a more chiastic and therefore polycontextural modeling than the uni-versalist approach proposed by the original UAM model[15].

Main differences

All those new approaches, like the Universal Actor Model, are based, at the end of the journey, on the strictly *algorithmic* and non-interactional concept of the Lambda Calculus with its simple abstraction procedures. Everything else, like reflection, distribution, interactivity, etc. is added secondarily and is a construct on the base of the primary calculus. Even if the Actor Model is surpassing the computational conception of the Lambda Calculus by definition, as it is postulated by Carl Hewitt, its *realization* is still fighting with the past, and topics like reflection, interaction and inconsistency (paraconsistency) are not genuinely incorporated and have to be added as an extension of the basic model. That is, some new generalizations and abstraction mechanism have to be added or supplemented on a higher level of the originary system based on the basic abstraction of naming and identification.

Diamond systems are from the very beginning distributed, mediated, reflectional and interactional, and based on *thematizations* instead of identification. All those features are at hand from the very beginning of modeling and computation, say mobile computing, at least on a *conceptual* level of modeling and implementation.

I also can't see any reason given by the reflectional UAM approach that could prevent it from being involved into all the known *conceptual problems* of computational reflection as studied by Brian Smith and Pattie Maes (meta-level architecture, meta-circularity, inspection, infinite regress).
This is by no means a failure of the conceptual designers involved but an inherent consequence of the general paradigm of thinking which is leading, i.e. limiting all those approaches. On a conceptual level all the possibilities of self-reflectional systems had been explored by philosophers long ago (Kant, Fichte, Schelling, Hegel, Heidegger, Ryle, Henrich, Tugendhat, Gunther).

Notes&References

1 Kenogrammatic systems are regarded simply as equivalence classes of semiotic systems.
This is the standard academic interpretation of Gunther's kenogrammatics. From that it follows, that the whole idea and apparatus of kenogrammatics is obsolete.
This opinion is based on ignorance in respect to the written texts, which are introducing kenogrammatics in a double gesture of philosophical interventions and mathematical inventions.
Because this field is still in its early stage of development, criticism is easy to apply. There is no serious academic carrier to do with it, hence deny its significance. But even with this ignorance, and based on tiny fragments of the trans-classical approach, some academic degrees had been achieved .

2 http://www.thinkartlab.com/pkl/ lola/AFOSR-Place-Valued-Logic.pdf

3 http://www.rudolf-matzka.de/dharma/semabs.rtf

4 http://www.logoi.com/notes/chinese_alphabet.html

5 http://www.thinkartlab.com/pkl/tm/MG-Buch.pdf

6 http://www.thinkartlab.com/pkl/media/SKIZZE-0.9.5-medium.pdf

7 http://www.thinkartlab.com/pkl/lola/Categories-Contextures.pdf

8 http://www-osl.cs.uiuc.edu/

9 **Hierarchy: primitive actors -> basic actors -> meta-actors**
It seems that the basic concept of the Actor Theory is not the Actor (event, message) but the *differences* between primitive/basic/meta, i.e., the architectonic distinction of different Actor types. The common trick of generalization/specification is not working. That is, a primitive actor is not simply a special case of a basic actor because the system of basic actors can not be defined consistently without the help of the primitive actors. Nor is it possible to define the meta-actor as a generalization of the common basic actors.
Thus, the differences in architectonics of the Actor Theory, up to now, is three-fold: primitive/basic, basic/meta and primitive/meta.
R. Kaehr, Actors, Objects, Contextures, Morphograms
http://www.thinkartlab.com/pkl/lola/Actors+Objects.pdf

10 **1.2.5 Universal Naming**
Since universal actors are mobile--their location can change arbitrarily--it is critical to provide a universal naming system that guarantees that references remain consistent upon migration.
Universal Actor Names (UAN) are identifiers that represent an actor during its life-time in a location-independent manner. An actor's UAN is mapped by a naming service into a Universal Actor Locator (UAL), which provides access to an actor in a specific location. When an actor migrates, its UAN remains the same, and the mapping to a new locator is updated in the naming system. Since universal actors refer to their peers by their name, references remain consistent upon migration.

1.2.5.1 Universal Actor Names
A Universal Actor Names (UAN) refers to an actor during its life-time in a location-independent manner. The main requirements on universal actor names are location-independence, worldwide uniqueness, human readability, and scalability. We use the Internet's Domain Name System (DNS) [Mockapetris, 1987] to hierarchically guarantee name uniqueness over the Internet in a scalable manner. More specifically, we use Uniform Resource Identifiers (URI) [Berners-Lee et al., 1998] to represent Universal Actor Names. This approach does not require actor names to have a specific naming context, since we build on unique Internet domain names.
The universal actor name for a sample address book actor is:
uan://wwc.yp.com/~smith/addressbook/
The protocol component in the name is uan. The DNS server name represents an actor's home. An optional port number represents the listening port of the naming service--by default 3030. The remaining name component, the relative UAN, is managed locally at the home name server to guarantee uniqueness.

1.2.5.2 Universal Actor Locators
An actor's UAN is mapped by a naming service into a Universal Actor Locator (UAL), which provides access to an actor in a specific location. For simplicity and consistency, we also use URIs to represent UALs. Two universal actor locators for the address book actor above are:
rmsp://wwc.yp.com/~smith/addressbook/
and
rmsp://smith.pda.com:4040/addressbook/
The protocol component in the locator is rmsp, which stands for the Remote Message Sending Protocol. The optional port number represents the listening port of the actor's current theater, or single-node run-time system--by default 4040. The remaining locator component, the relative UAL is managed locally at the theater to guarantee uniqueness.
While the address book actor can migrate from the user's laptop to her personal digital assistant (PDA), or cellular phone; the actor's UAN remains the same, and only the actor's locator changes.
The naming service is in charge of keeping track of the actor's current locator.

1.2.5.3 Universal Actor Naming Protocol
When an actor migrates, its UAN remains the same, and the mapping to a new locator is updated in the naming system.
The Universal Actor Naming Protocol (UANP) defines the communication between an actor's theater and an actor's home, during its life-time: creation and initial binding, migration, and garbage collection.
UANP is a text-based protocol resembling HTTP with methods to create a UAN to UAL mapping, to retrieve a UAL given the UAN, to update a UAN's UAL, and to delete the mapping from the naming system.
Gul Agha and Carlos Varela. Worldwide Computing Middleware. In M. Singh, editor, Practical Handbook on Internet Computing. CRC Press, 2004. http://wcl.cs.rpi.edu/papers/chmiddleware.pdf

11 http://www.lix.polytechnique.fr/Labo/Robin.Milner/bigraphs-tutorial.pdf

12 http://www.thinkartlab.com/pkl/lola/Interactivity.pdf

13 http://www.thinkartlab.com/pkl/media/Diamond_Web2.0/Diamond_Web2.0.html

14 http://osl.cs.uiuc.edu/docs/firstpaper/final.ps

15 http://yangtze.cs.uiuc.edu/Theses/varela-phd.pdf