6-13-2008

METAPATTERN: Information Modeling as Enneadic Dynamics

Pieter Wisse
University of Amsterdam, pieter@wisse.cc

芽|Sprouts Working Papers on Information Systems ISSN 1535-6078

METAPATTERN: Information Modeling as Enneadic Dynamics

Pieter Wisse
University of Amsterdam, The Netherlands

Abstract
An object may exhibit multiple behaviors. Every behavior is unambiguously tied in with a particular situation. With a context representing a situation and with signature as an object's bare identity, through a number of signature instances an information model represents an object in multiple contexts. This concept of context is especially characteristic of the metapattern approach to information modeling. Context is a recursive function of both signature and relationship between two signatures at adjacent levels in the model. It suggests a development of the classic semiotic triad of C.S. Peirce, at first onto a system of hexadic and next onto a system of enneadic dynamics. The design of the ennead, like the original triad, equals semiotics with ontology and epistemology. The increase in modeling variety the metapattern offers is sketched through a fictional assignment. To gain a familiar perspective, basic assumptions underlying entity-atribute-relationship modeling and tradititional object orientation are revisited. Next, the metapattern's assumptions as they relate to multiple, recursive contexts are presented.

Keywords: metapattern, context, information modeling, semiotics

Permanent URL: http://sprouts.aisnet.org/1-4

Reference: Wisse, P. (2001). "METAPATTERN: Information Modeling as Enneadic Dynamics," University of Amsterdam, Netherlands . *Sprouts: Working Papers on Information Systems*, 1(4). http://sprouts.aisnet.org/1-4

1. Introduction

The metapattern is a technique for meta-information analysis and modeling. It emphasizes reusability. It adds precision through the combination of a finely grained concept of time stamping and a recursive, simple but formal concept of context. The metapattern is particularly valuable for aligning complex and variable requirements, even across a multitude of organizations with different processes. The concepts of *context* and *time* are critically important in these models, allowing for their adjustment to time-induced and/or situational changes that the model must account for to maintain its integrity.

The metapattern's basic concepts, their structure, a comparison with 'traditional' object orientation, and a host of practical modeling cases are presented in *Metapattern: context and time in information models* (Wisse 2001). Written for an audience of professionals rather than scientists, *Metapattern* deliberately passes over ontological considerations. These are taken up in *Semiosis & Sign Exchange: conceptual grounds of business information modeling* (Wisse forthcoming). This paper is derived from its fourth chapter. Concentrating on context, the metapattern is explained and applied to expand an ontology annex epistemology annex semiotics called subjective situationism. Concepts appear in a *different* configuration; the meanings of some familiar terms change accordingly.

2. Hexad: grounded expansion of the triad

Does Peirce's semiotic triad reflect his own concept of sign? It must surely be the most cited definition of semiotics that a sign (Peirce 1897)[1] "is something which stands to somebody for something in some respect or capacity." What the triad doesn't account for is the qualifier "in some respect or capacity." Peirce himself provides only minimal clues. He mentions that a sign (Peirce 1897) "stands for that object, not in all respects, but in reference to a sort of idea, which I have sometimes called the *ground* of the [sign]."

Instead of a single ground, the metapattern includes three different grounds. Each corresponds to an element from the original triad (sign, object, interpretant). The result is a hexad, as shown in Figure 1. More specifically, a context acts as ground for a sign, a background interpretant as ground for a foreground interpretant, and a situation as a ground for an object. The original three triadic elements of Peirce reappear as dimensions along each of which two more finely-grained concepts are positioned.

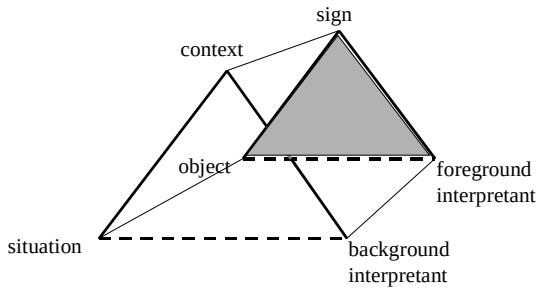

Figure 1.
Two connected triads make a hexad.

Then, isn't context of a sign nature, too? It is. Sign therefore consists of both context and … sign. However, without explicitly shifting levels of abstraction, it is impossible that a particular element (sign) is equal to the set (sign) when another element (context) is present which is by definition[2] non-empty. This ambiguity is depicted in Figure 2.

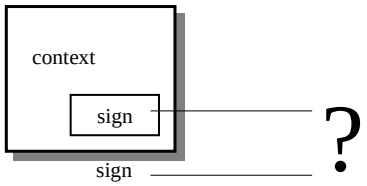

Figure 2.
Sign as an ambiguous concept.

For precision, either the set-level concept or the element-level concept must be renamed. I retain *sign* at the level of the set. *Signature* is the new name for what I originally called sign at the elementary level. As Figure 3 shows, a sign consists of a signature in a context.

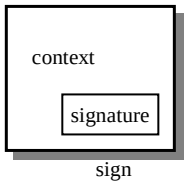

Figure 3.
Introducing the label of signature.

This change of terminology is so important that it requires Figure 4 as an update of Figure 1. Please note that the *only* change is from sign to signature as the name for one of the hexad's elements.

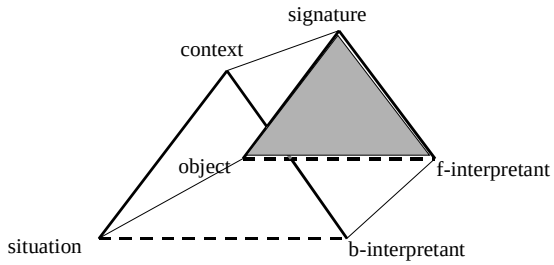

Figure 4.
Signature: update in terminology.

The hexad suggests what counts as a comprehensively articulated sign, or model. The modeler develops a signature representing the object. He also develops a context representing the situation, and next includes the signature in the context so that it may stand for the object's occurrence in the situation. At the basis lies the correspondence between f-interpretant and object, and b-interpretant and situation, respectively. Figure 5 shows both the correspondences and the containments of the concepts from the hexad.

The minimal difference that a modeler interprets as (his) reality is that between a situation and an enveloped object. It follows that the minimal difference that s/he constructs onto a model should be that between a context enveloping a signature. The metapattern is a technique for consistently differentiating between signature and context. It is nontrivial because contexts and signatures are not absolutes. Neither are their relationships fixed. A model must therefore also support representation of shifting points of view. The metapattern, especially as a technique for model visualization, is about *variously* directing attention toward what may be taken as signature, and what as context, in a sign. In fact, a third concept along the sign dimension is that of intext. It will be introduced later on. Adding corresponding elements along the two other semiotic dimensions, the hexad transforms into an ennead.

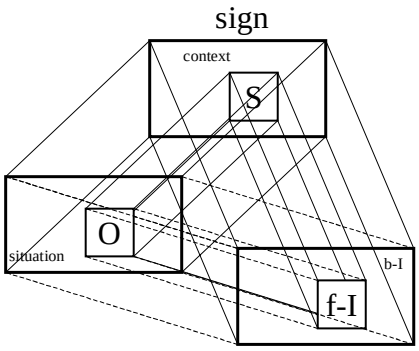

Figure 5.
Hexadic concepts: correspondences and containments.

3. Traditional modeling practice

The metapattern's principles are outlined by presenting a fictional, simple case study. In this paragraph, I first show what assumptions guide a modeler who applies the frames of reference of relationships between entities with attributes and/or object relationships. Entity-attribute-relationship modeling (EAR) and object orientation (OO) share the ontology of absolute, independently existing objects.[3] Then, the next paragraph sketches the metapattern.

Suppose, one day, a modeler goes to work. There, he learns his new assigment. But all he hears at the initial briefing is that the prospective information system "has to do with people."

Were he to take the traditional approach to modeling, he would probably conclude from "people" that some number of persons is involved. His next move would likely be that, at least for the moment, differences between individual persons are not important. When he concentrates on their similarities, he moves up a level of abstraction. It is person-as-type that he is now interested in to learn more about. As a consequence of this abstraction, person-as-instance is seen, not as an element that contributes to the extensional definition of the person set, but as a possibility that can be materialized from the set's intensional definition.

Still taking the traditional approach to information modeling, the person at type level is then fitted with relevant properties. The key question at this stage is, of course: What is relevant? It all depends. What purposes is the information about persons going to serve? As the modeler was not told about any purpose, he has a choice between [a] doing nothing, [b] inventing one or more purposes himself, or [c] trying to find out about them from (other) stakeholders. With the exception of [a], the modeler will attempt to specify properties to suit.

Is the information system going to support, for example, integrated management of magazine subscriptions? Suppose it will have to. He attributes magazine type as a property of person type. Through instantiation, a particular person – John, for example – can be subscribed to a particular magazine – say, *Business Semiotics Weekly*. What possibilities may be generated from type-oriented modeling are outlined in Figure 6.

The relationship between the types of person and types of magazine precribes what relationship is possible, allowed, etcetera, at the instance level. Because any person holds the potential for subscribing to any magazine, the actuality may be generated for John to have a subscription to *Business Semiotics Weekly*.

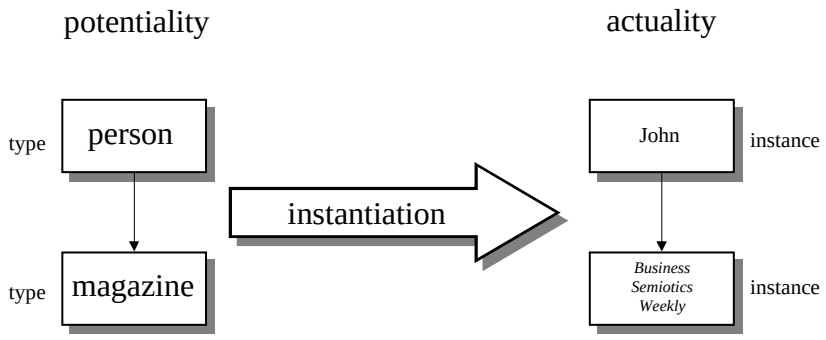

Figure 6.

Modeling with type-level priority.

Information models are traditionally almost without exception at the level of types. It arises to a large extent from the popular languages for computer programming and database management systems. It has been built into them to support types. What is often less understood, is that those are computer-oriented types. They match the internal, limited variety of digital-machine information types. Nearly always, those are not the types that are directly suited for a human subject's classification, i.e., to reflect purposeful order in his reality.

It is impossible to completely escape from the worldview tools incorporate. A responsible modeler should at least be aware of inevitable bias. So, modeling right away at type level is actually jumping to a conclusion. There is the risk of failing to recognize real problems when the relevant variety of reality is missed. Strictly intensional modeling might be insufficient to support requisite variety. The metapattern holds that reality is made up of *particular* situational objects. Of course, instances may be grouped into classes, or sets. Extension is what *essentially* constitutes a type; intension can be a convenient shorthand, with all the risks of reduction.

For the remainder of my treatment of traditional information modeling, here in this paragraph, the orientation at type level is implied. What other stakeholders usually experience from the efforts of the traditional modeler are extremely condensed schemata such as Figure 7. Its simplicity, that much should be clear, is misleading.

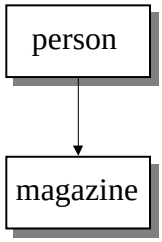

Figure 7.

Model with implicit type orientation.

For the sake of presentation, suppose this is how far the traditional modeler has proceeded. He will then, I assume, wonder about the quantative aspect of the relationship between person and magazine. How many – types of – magazines can a person subscribe to? Just one? An unlimited number? Is information about the person registered in advance of his very first subscription? Does personal information continue to be available when no subscription remains?
And what about numbers from the perspective of the magazine type? How many persons can subscribe to it? Etcetera. For a healthy publishing company the modeler specifies a many-to-many relationship (see Figure 8). Those numbers are also called the cardinality.

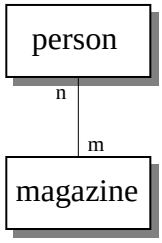

Figure 8.
Specifying cardinality at type level.

A many-to-many relationship lacks sufficient precision. It does not support unambiguous selection of information, for example, on the particular magazines subscribed to by a particular person. The proven solution for disambiguation is the introduction – again at the type level, here – into the model of a third entity/object. In this case, that object is subscription. Keeping the model general, i.e., without specifying when the presence of an object instance of one type is a precondition for the production of an instance of another type,[4] Figure 9 presents the solution to the ambiguity problem.

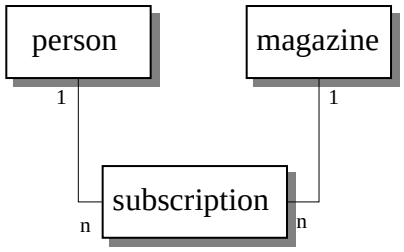

Figure 9.
Disambiguing the model: substituting one-to-many relationships for many-to-many relationship.

It usually happens only when programming, or related activities of organizational change, etcetera, is well on its costly way that flaws in conceptual models are discovered. It is then established that not only private persons but organizations, too, subscribe to magazines. This would not be a problem when

they are not different, at all, from the *perspective* of subscription management. All it takes, then, is to rename person. Subscriber, for example, will come to mean any party.

There are often valid reasons for maintaining – the possibility of – differences between, in this case, person and organization. Where the subscription perspective is dominant, person and organization will be established as mutually exclusive subtypes of subsciber. The modeler adjusts his model as shown in Figure 10. The diamond-shaped symbol indicates a choice between mutually exclusive results. What needs to be controlled as a result of this alternative is the choice between person and organization as a property of subscriber.

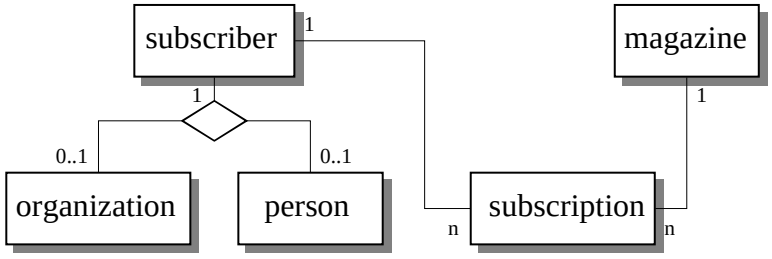

Figure 10.

Subscriber as the predominant type: organization and person as subtypes.

If the difference between person and organization is of primary interest, the traditional modeler has to resort to so-called subtypes. This will actually always be the preferred solution when person and organization are not only different, but their respective subscriptions as well. See Figure 11.

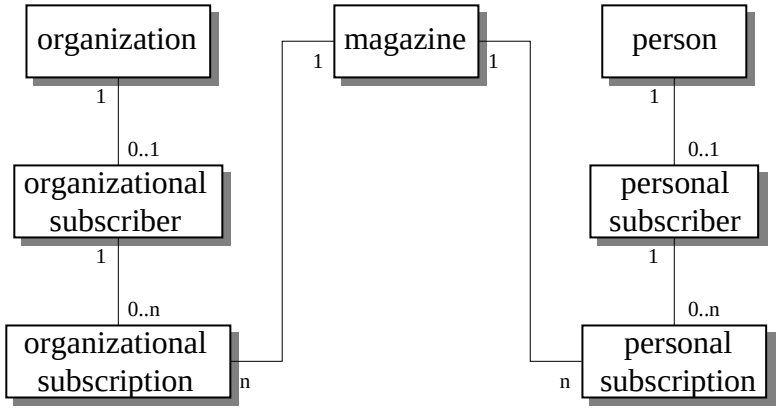

Figure 11.

Different subscription types for organization and person, respectively.

Strictly regarding subscription management, often no differences exist between person and organization. The orientation at subtyping, however, makes it impossible to escape from proliferation of differences inside the model where *no* such differences exist in reality.

4. Modeling with a difference

Another modeler, one familiar with the metapattern, responds differently to the same assignment. The metapattern-equiped modeler will, first of all, try to gain an understanding about what *particular situation* is relevant. She will concentrate, for example, on: John having a subscription to *Business Semiotics Weekly*. What she is looking for is a clue as to how she can start to objectify the situation-as-instance, rather than object instance(s) residing in it. She might call the particular *situation*: subscribership.

She can relax on a fixed object orientation because — the metapattern determines that —situation and object are relative concepts (Wisse 2001). A particular situation is also object in another, higher-level situation. Likewise, an object can act as situation in which another, lower-level object resides. Situation, then, is a recursive function of object and relationship.

Aware of the opportunity for upward recursion, she will next investigate whether subscribership in its capacity as object leads to recognition of another, encompassing situation instance. Suppose she cannot, at such short notice, think of any. Neither are there any stakeholders around to suggest such situations with broader relevance. She will therefore put a firm limit to upward recursion. This is easily accomplished in a visual model by drawing a thick, continuous line. That line represents the horizon of her reality with regard to the prospective information system. It provides her the base in (and of) the model.

The situation of subscibership is now included in the model as if a relationship originates from reality's horizon. From the base line, a thinner line is drawn, and a name for the situation is added. See Figure 12.

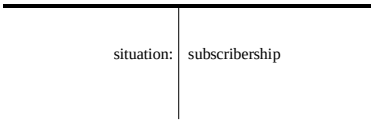

Figure 12.

The relevant horizon of reality, drawn at the top; a situation instance specified through a relationship with the horizon.

Only *after* a situation has been specified does she give closer attention to persons. Actually, what she does is to include John – hypothesized or not – as an individual, first. Her next sketch, therefore, looks like Figure 13.

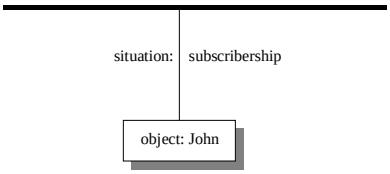

Figure 13.
Placing an object instance inside a situation instance.

What this procedure especially supports is to help the modeler focus her attention as much as possible on how a *particular* object exists within a particular situation. What is relevant for the modeler about that existence is differential *behavior*. Because the situation has been presumed to govern behavior, it is John *as a subscriber only* that the modeler needs to consider at present.

In actual practice, the modeler iterates, of course. Behavior is the joint result of object and situation. So, particular behavior is their relationship. Practically, she has to start her model somewhere. It is by assuming greater importance for situation. Within this assumption, it is reasonable to speak of situation governing behavior.

Whatever *other behaviors* by John, or roles he occupies, may be relevant in the past, now, or at any time in the future, should also be understood within their corresponding, and most likely *different*, situations. Anything applying to other situations, therefore, is *not* relevant within the current focus of the particular situation. It is this concept of situation that characteristically sets the metapattern apart from traditional object orientation (and provides it with advantages over OO; Wisse 2001). Compared to an object that (only) exists absolutely, an object believed to exist in a multitude a different situations can unambiguously be modeled – to be equiped – with corresponding behavioral multiplicity.

For information modeling, it is useful to stretch the meaning of behavior. And for all purposes, behavior and role may be considered synonyms. They are generic terms for the set of properties, both static and dynamic. For static properties are really not fundamentally different from dynamic properties. Static ones just have the absence of change as their boundary condition.

The modeler proceeds to investigate John's behavior as a subscriber. Soon enough, she will also turn to abstraction. For the prospective information system, is it really necessary to differentiate behaviors of separate individuals? What the metapattern urges is that individual existence must always be honored. But assuming for now that John's behavior does not differ in kind from the behavior of all other *subscribers*, she may substitute 'person in general' for John. The number of persons who can be subcribers is now added to the model. This is their cardinality. When she doesn't want to occupy herself at this early stage with detailing any properties of subscribers, a text balloon suggests that one or more of such properties do exist. Figure 14 presents what she has modeled so far.

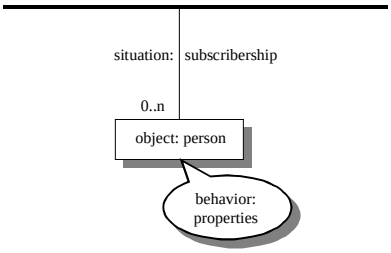

Figure 14.

Abstraction from instances to type; specifying cardinality; suggesting behavior annex properties.

The focus on situation leads the modeler to question whether only subscribership is relevant for persons. Suppose that John needs to be registered as a member, too. She learns that his membership is with the Global Semiotic Society. However, for the time being she chooses to concentrate on John as far as instances are concerned. For the sake her model's compactness, rather than from type as a dangerous principle, she assumes that all members will be registered by the same types of properties as their behavior. This abstraction from member instances to the behavioral type of membership is shown in Figure 15.

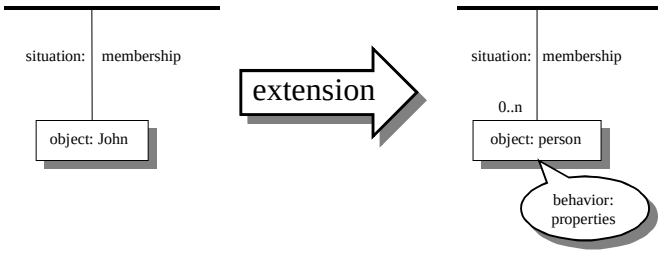

Figure 15.

Hypothesizing similar behavior for all members leads to recognition of a generally applicable situation of membership.

Back again at the level of the individual John, she merges her previous models. See Figure 16.

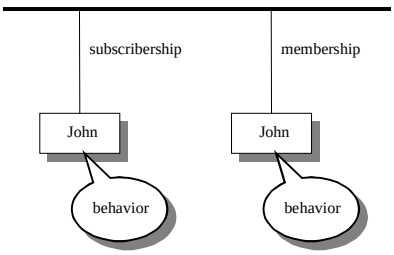

Figure 16.

John with different behaviors in corresponding situations.

In all its simplicity, Figure 16 shows exactly what the task of the information modeler is. She designs an appropriate balance between sameness and difference. In this case, sameness is expressed by identifying a single object. That is the role of John in her model. But John is everywhere circumscribed by difference. First of all, he appears in different situations, i.e., in subscribership and membership, respectively. Secondly, he is attributed with different behaviors. In general, *the difference of situation establishes for the sameness of object the difference of behavior*.

5. Sign in the ennead: context, signature and intext

Now the modeler has reached the point where she has modeled – actually, she has made preparations to model – two behaviors of John, one as a subscriber, and the other as a member. Suppose those behaviors are completely disjunct, that is, no overlap of properties exists. This raises the question about what counts as minimal information required to connect John's behavior to the situation.

At the point where John enters the situation even the barest *identification* is sufficient. The metapattern separately establishes this minimal point. That, and *only* that, is John's signature.

But what in the model, then, informs about John's behavior in a particular situation? This is named, as a pendant of context: intext. As Figure 17 indicates, the model-as-sign is a variable configuration, not of two, but of three concepts: context, signature, and intext..

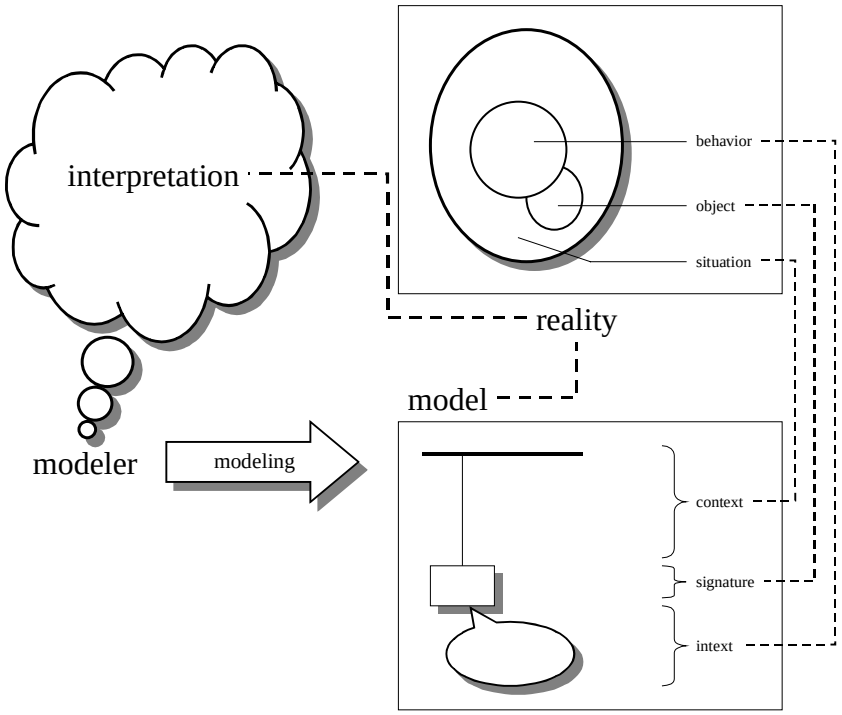

Figure 17.

Reframing the concept of signature: making room for intext to represent behavior.

The repositioned meaning of signature underlines that an identification does not 'stand for' an object in any other way than providing a *minimal* reference to it. What *really* does characterize an object is its behavior. In fact, given a particular situation, that behavior *is* the situational object.

The articulation of both a sign and reality each into three, rather than two components, should be reflected in the model of semiosis. It makes sense to assume that a structural equivalent of situational object and – its – signature is present along the dimension of interpretation. I call it *focus*. What results is an ennead, replacing the hexad. This development from six to nine constituting elements is shown in Figure 18.

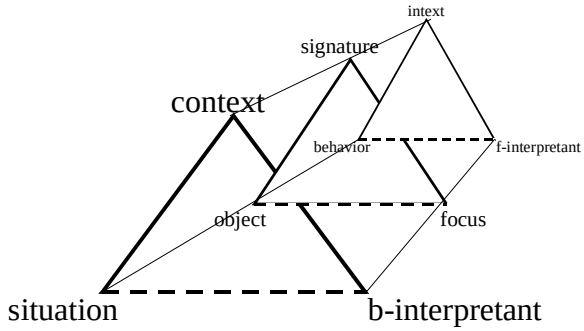

Figure 18.

Enneadic model of a step in processes of sign use.

The ennead is a powerful interdisciplinary device. It retains the original elements of Peirce's triad as dimensions. Along each dimension, concepts are now arranged to create formal structure in models. Correspondence between phenomena along the different dimensions can be established in detail. Applying the metapattern, from the sign-as-model it is possible to infer more rationally about both the configuration of interpretants and the configuration of reality (where the latter is of course inferred from the interpretive structure which the sign mediates). Independently from Peirce, Voloshinov remarks (Voloshinov 1929):[5] "[T]he inner psyche is not analyzable as a thing but can only be understood and interpreted as a sign."

The radical conclusion from the orientation at situational behavior is that an object's identification is *behaviorally meaningless*. The modeler does not have to explicitly include something like an original signature in all her models. Essentially a privileged situation may implied. It serves the only purpose of guaranteeing sameness or, its equivalent, persistent identity across (other) situations. Being a situation in its own right, when included in a model it is represented by a seperate context. Made explicit or not, its role is to authenticate an object's identity in other situations by establishing the signature in other contexts.

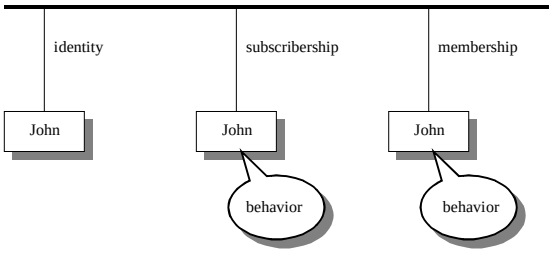

Figure 19.
A separate identity context.

This touches upon the reasons why I introduced, in § 2, the term signature. A signature itself does not carry information except for leading to an intext as a particular context directs, vice versa. This way, it stands for an object where it exhibits behavior in a situation. Figure 19 shows the model accordingly expanded.

Literally through the concept of signature, context and intext become concepts that are (more) independent from each other. For how instances of context relate to instances of intext can always change around signatures. This explains the modeling power of the metapattern (Wisse 2001).

As a consequence of its behavioral emptiness, an object in its identity situation is considered propertyless. Therefore, no intext is shown in the model. Again, an identity serves to preserve an object's sameness across situations and its corresponding behavioral differences. And when an object is established in a particular (other) situation, in the model it must draw its signature from its 'central' identity. When that object has not yet been observed in *any* situation, its identity must be established as a prerequisite for entry in any (other) situation. Obviously, when an object's existence in the past, present and future is no longer considered relevant in any (other) situation, its otherwise empty identity is also no longer required.

Through lateral connections between signatures, an object's existence in one situation may be derived from its existence in another situation. As Figure 20 shows, such relationships are included in the model as curved, broken lines with their arrow pointing at the 'originating' signature. Not shown are secondary derivations, but there is no reason why an additional signature instance of John should not be drawn from his, for example, membership signature, etcetera. As a matter of metapattern principle, directly or indirectly, a signature is always derived from its identity. As an axiomatic value, the identity's signature is considered equal to the identity.

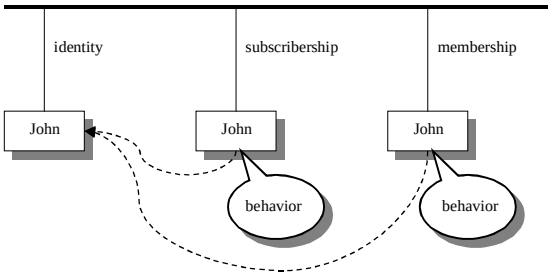

Figure 20.
An object's continuity across situations: making derivative relationships explicit.

It is too cumbersome to include these fundamental relationships to an object's central identity in all information models. When they are absent from a particular model, they should be presupposed. What the modeler continues to concentrate on, is situational behavior. She will model situations onto contexts with the purpose to eliminate behavioral duplication. Disjunct behavior defines situations as disjunct.
Initially, for example, John as a subscriber and John as a member both require his address in the respective intexts. It will sound contrived at first, but the additional situation of, say, personship, eliminates the duplication. As it is clear that John as a natural person exists before he can ever be considered a subscriber and/or a member, derivations of signature are changed as shown in Figure 21. The privileged identity situation is left out for the sake of compactness; it resides in the background..

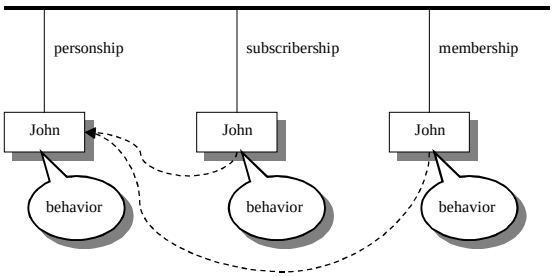

Figure 21.
Elimination of duplication in behavior results in situations that are by definition disjunct.

There is more to be said about an object's identification being behaviorally meaningless. And about why the object in its identity situation is essentially propertyless. Especially the *name* 'John' should not be taken as the individual's signature, not even where his personship is involved. Rather, any name is better considered a property. The relationships leading from the objects in other situations to personhood guarantee that John-as-name can be made available there, too.
The radical nature of signatures, only serving to connect context to intext, allows the models to be presented more simply. In Figure 22, John-as-a-name is now a property of an individual, no longer his

signature. This illustration also does away with the unusual naming of situations. They are renamed according to the *role* the object plays in that particular situation.

In computerized information systems, the 'machine' can easily provide a single identification value across the different signature instances for a particular object. A user may never notice how an object's sameness is organized through actual information. All he experiences are meaningful situations from his perspective(s) and, within every situation, meaningful properties of any object.

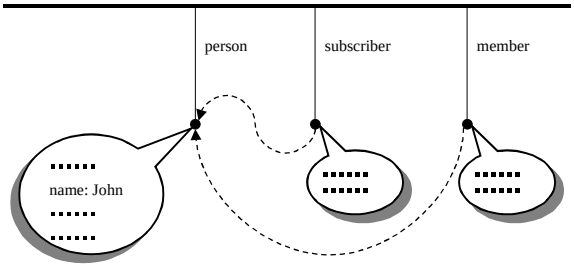

Figure 22.
An object's common name as a property in an appropriate situation.

The modeler can, again, choose for abstraction. She then has to include, as in Figure 23, her idea about the number of objects that may exist within every situation. Text balloons remain because they alert to unfinished modeling. Relationships shown for derivations provide additional information about the number of objects in a situation. As the dotted arrows for example make clear, it is impossible for a member to be present without a natural person as its 'origin.'

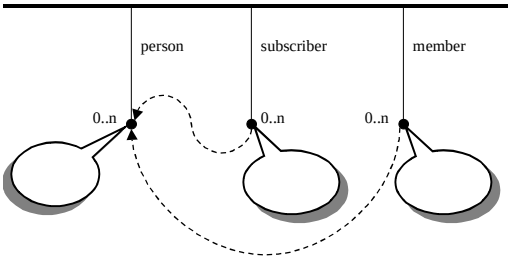

Figure 23.
Again, moving from object instances to situation-determined types of behavior.

6. Relative configurations

The metapattern's principles for multiple contexts have now been explained. Applying them, the fictional modeler produces a model introducing magazine and subscription. See Figure 24.

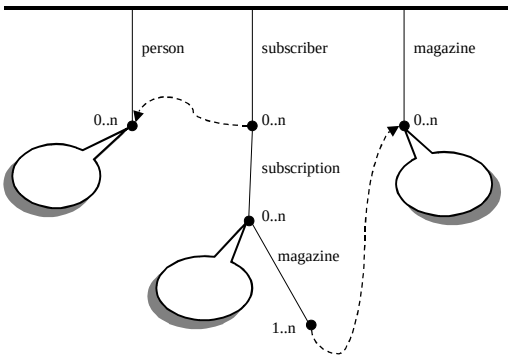

Figure 24.
After mastering the metapattern's basic technique, models can be elaborated.

A provision is still missing for organizations to subscribe to one or more magazines while maintaining the difference between persons and organizations. The radical notion of identity in reality, and correspondingly of signature in the model, makes for a simple solution. Assuming that a person-as-subscriber is not different from an organization-as-subscriber, the separate situation of subscribership is useful. It 'contains' *by definition* only one kind of behaving object. They are subscribers. How those objects behave in other situations is of no concern *from the perspective of subscribership*. So, whatever behaviorial differences are 'supported' by other situations does not matter. What is considered a car elsewhere, for example, may also entertain a magazine subscription.
To indicate that a choice exists at the level of individual signatures, and how they fundamentally connect to sameness in different situations, in Figure 25, a pertinent symbol is added to the arrows of derivation.

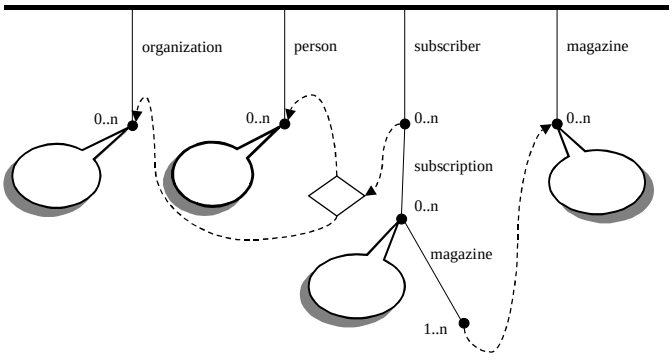

Figure 25.
Objects with heterogeneous behavior elsewhere are all placed in the same situation when their behavior is homogeneous within that particular situation.

The onorthodox idea of making cars eligible to magazine subscriptions is, by the way, not shown above. But modeling it is simple enough. Situations eliminate the need for this kind of subtyping.

Because these particular situations are juxtaposed, so are their types. For by definition, situation is the type for all objects behaving in it.

The modeling example has now progressed far enough to explain that context, signature, and intext are not fixed categories. Above, it already says that situation is a recursive function of object and relationship. This is actually inferred from the metapattern's concept of context, i.e., as a recursive function of node/signature and relationship. Take, for example, the point of view of the signature as indicated in Figure 26.

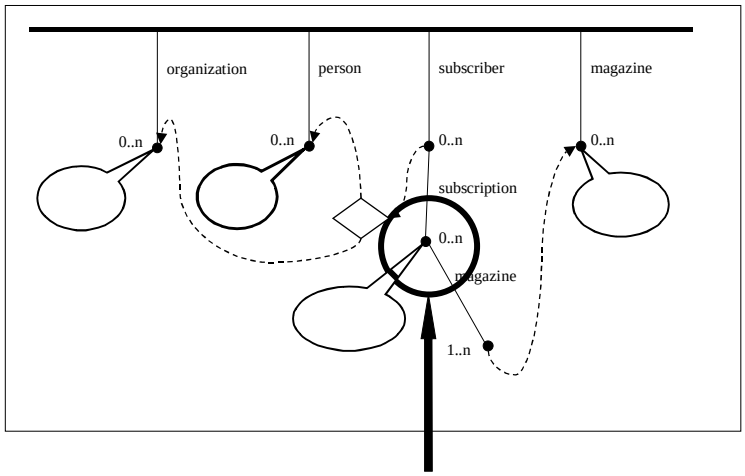

Figure 26.

A metapattern-based model invites the sign user to choose focus.

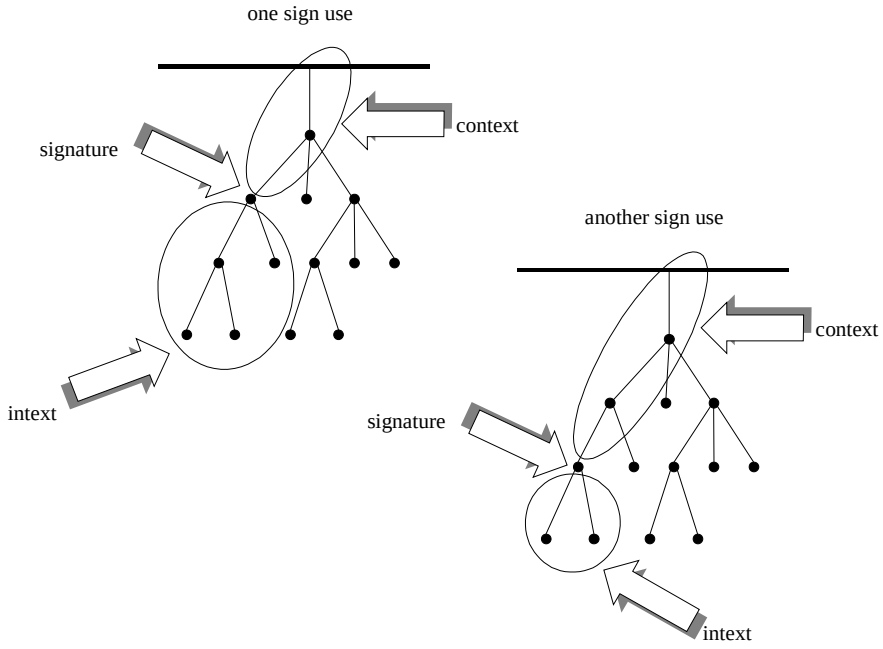

Figure 27.

Support of different interpretations (also read: sign uses).

It is precisely a signature that supports a focus. The *experience* of a signature or a point of view *is* a focus, even. Starting from a particular signature, its context is the specification of the situation. Its intext is all what specifies behavior of the situational object. Every change of point of view/signature changes the context and the intext, too. The metapattern thus supports a large variety within compact models. Figure 27 suggests two different 'readings' from the same model. Every interpretation is driven by a focus. This establishes the signature, and subsequently the related context and intext. A different focus results in the experience of a different signature, etcetera.

It may be argued that context is not just the narrow definition of a situation. For a situation does not exist in isolation. It exists as an element in a *system* of situations. As such, one situation is determined by all other situations; similarly, that situation participates in the establishment of other situations. The *narrow* context, then, is just the path leading from the overall perspective to the signature in question. And the *wide* context is everything the model presents except for that signature and its intext. This view is shown in Figure 28.

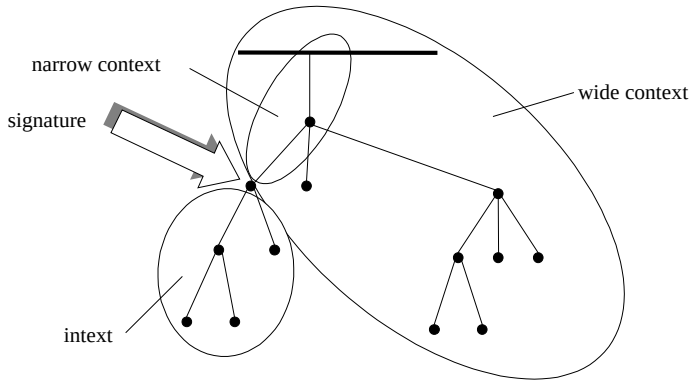

Figure 28.
Narrow context is linear, wide context is structuralist.

Offering a taste of additional possibilities, the fictional case study is extended through a final step to show how the modeler can proceed. It was suggested that persons need to be registered as members, too. Suppose that the modeler hears a person can only subcribe to particular magazines in his capacity as a member. She can easily adjust the model to simultaneously accommodate different situations of subscribership. It should be clear that the model of Figure 29 also covers organizations receiving a magazine as part of some membership. A subscriber can be either a person, an organization or a member. And a member is either a person or an organization.

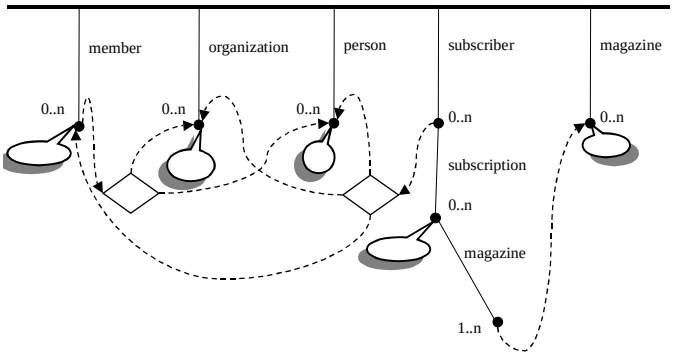

Figure 29.
Modeling a network of unambiguous behaviors.

7. Final remarks

Concepts labeled context and/or situation are not at all new to for example significs (also read: semiotics; Mannoury 1925), linguistics (Bréal 1897, Voloshinov 1929, Bühler 1934), logic (Dewey 1938; Barwise and Perry 1983), and information modeling (Sowa 1984, 2000; Kilov 1999). Especially social psychology maintains a strong tradition of situational explanation of behavior (Mead 1934). The novelty that underlies the metapattern consists of combining both situation and context as recursive functions in an encompassing, interdisciplinary semiotic framework of enneadic dynamics. What results are, among others, *new concepts* of situation and context. Applying definitions from other frameworks fails to provide the proper focus.

The metapattern has more characteristics than can be described here. For example, time receives pervasive treatment. Every node accommodates time. For one thing, it serves to integrate aspects known from data warehousing into operational information systems. Audit trails are also intrinsic. Those features are documented elsewhere (Wisse 2001). This paper outlines the metapattern as a technique for modeling both actual and planned reality in general. Its essential characteristics are that context is a variable *within* the scope of information models and that context as a recursive function yields highly compact models with large variety.

Notes

1. I've taken both quotations appearing in this paragraph from Peirce's essay 'Logic as semiotic' as compiled by J. Buchler in the collection *Philosophical writings of Peirce*. The year follows Buchler's suggestion for the time at which Peirce has written the particular sentences.
2. Whenever I write "by definition" it is actually more instructive to read it as: by situationism as ontology.
3. I have not supplied references to literature about EAR and OO. A reader interested in more details of how the metapattern compares can find references to EAR and OO literature in my books *Aspecten en Fasen* (1991), *Informatiekundige ontwerpleer* (1999), and *Metapattern* (2001). *Metapattern* also engages in a comparison with the traditional object orientation to conceptual information modeling. There, the metapattern's advantages over OO are explained in detail guided by a review of J.J. Odell's collection of OO-modeling essays *Advanced Object-Oriented Analysis & Design Using UML* (1998).
4. It should normally be impossible, clearly, to record a subscription instance without both the person instance involved and the magazine instance requested being already present.
What I did not elucidate in the main text is that magazine instance is meant here as an instance of a magazine type. It is, of course, possible, to consider magazine issues, and copies of every issue. I did not want to burden the fictional case study with such elaborations. However, on the basis of this note, the reader is invited to do so for himself.
5. Voloshinov also published about the psychoanalytic theory of Freud.

Literature

Barwise, K.J. and J. Perry (1983) *Situations and Attitudes*, reissue edition 1999, CLSI, Stanford.

Bréal, M.J.A. (1897) *Semantics: Studies in the Science of Meaning*, translation from the French 1964, Dover, New York.

Buchler, J. (1955) (ed.) *Philosophical writings of Peirce*, Dover, New York.

Bühler, K. (1934) *Sprachtheorie: Die Darstellungsfunktion der Sprache*, edition 1965, Gustav Fischer, Stuttgart.

Dewey, J. (1938) *Logic: The Theory of Inquiry*, edition 1960, Holt, Rinehart and Winston, New York.

Kilov, H. (1999) *Business Specifications: The Key to Successful Software Engineering*, Prentice Hall, Upper Saddle River.

Mannoury, G. (1925) *Mathesis en Mystiek: een signifiese studie van kommunisties standpunt*, Wereldbibliotheek, Amsterdam.

Mead, G.H. (1934) *Self, Mind, & Society*, edition 1974, University of Chicago Press, Chicago.

Odell, J.J. (1998) *Advanced Object-Oriented Analysis & Design Using UML*, Cambridge University Press/SIGS, Cambridge.

Peirce, C.S. (1893-1910) Logic as semiotic, *Philosophical writings of C.S. Peirce* (J. Buchler, ed.).

Sowa, J.F. (1984) *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, Reading.

————. (2000) *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks/Cole, Pacific Grove.

Voloshinov, V.N. (1929) *Marxism and the Philosophy of Language*, translation from the Russian 1973, Seminar Press, New York.

Wisse, P.E. (1991) *Aspecten en Fasen*, Information Dynamics, Voorburg.

————. (1999) *Informatiekundige ontwerpleer*, Ten Hagen & Stam, The Hague.

————. (2001) *Metapattern: context and time in information models*, Addison-Wesley, Boston.

————. (forthcoming) *Semiosis & Sign Exchange: conceptual grounds of business information modeling*.

METAPATTERN:
information modeling as enneadic dynamics

Pieter Wisse

ABSTRACT: An object may exhibit multiple behaviors. Every behavior is unambiguously tied in with a particular situation. With a context representing a situation and with signature as an object's bare identity, through a number of signature instances an information model represents an object in multiple contexts. This concept of context is especially characteristic of the metapattern approach to information modeling. Context is a recursive function of both signature and relationship between two signatures at adjacent levels in the model. It suggests a development of the classic semiotic triad of C.S. Peirce, at first onto a system of hexadic and next onto a system of enneadic dynamics. The design of the ennead, like the original triad, equals semiotics with ontology and epistemology. The increase in modeling variety the metapattern offers is sketched through a fictional assignment. To gain a familiar perspective, basic assumptions underlying entity-atribute-relationship modeling and tradititional object orientation are revisited. Next, the metapattern's assumptions as they relate to multiple, recursive contexts are presented.

KEY WORDS AND PHRASES: metapattern, context, information modeling, semiotics

Contents

芽|Sprouts

Working Papers on Information Systems | ISSN 1535-6078

Office:
Sprouts
University of Amsterdam
Roetersstraat 11, Room E 2.74
1018 WB Amsterdam, Netherlands
Email: admin@sprouts.aisnet.org

芽|Sprouts