How Science Moves

Interdisciplinary Import in Human-Centered Informatics.

Janus Hove Jørgensen
Speciale, Humanistisk Datalogi
Juni 2008

Vejleder: Henrik Schärfe

How Science Moves – Interdisciplinary Import in Human-Centered Informatics.

Aalborg Universitet, juni 2008.

191.697 typeenheder.

Speciale, Humanistisk Datalogi.

Vejleder: Henrik Schärfe

Janus Hove Jørgensen

Table of Contents

ABSTRACT

Despite many attempts, the concept interdisciplinarity seems difficult to define, at least in a linguistic form. This has caused for some confusion of concepts and a lack of clarity regarding method in interdisciplinary science. However, tendencies toward e.g. problem-orientation, holistic approaches as wells as the emergence of new technologies make it relevant to examine how interdisciplinarity unfolds. Human-Centered Informatics (Humanistisk Datalogi) is a good and typical example of a field in which interdisciplinarity is a basic term.

Drawing upon the semiotics of C.S. Peirce, I will present a conceptialisation of interdisciplinarity within Human-Centered Informatics. This conceptualisation is expressed through dividing science into a trichotomy of 'Domain', corresponding to Peirces notion of firstness, 'Analysis', corresponding to secondness, and 'Theory' corresponding to thirdness.

With Richmonds [2005, 2006] idea of |>* (trichonic) and his vectorial analysis as a basis, interdisciplinarity is conceptualised as movements within this scientific triad. In this process, elements from other sciences are imported in the three different universes 'Domain', 'Analysis' and 'Theory'. I establish that this happens through some form of likeness in any of the three universes, and provide a simplified definition of what likeness means in this context. This conceptualisation of interdisciplinary science is a metaphor, according to which scientific elements 'move' from other sciences into Human-Centered Informatics.

From this idea, I develop and present a model for interdisciplinarity, consisting of 6 different triadic operations which may come into play in interdisciplinary science. I provide examples of these 6 operations by applying them in an analysis in which I draw upon a broad selection of literature from the field of Human-Centered Informatics.

I then proceed to derive the insights provided by the model, along these main lines: (1) The nature of the field Human-Centered Informatics, (2) The principle of tranfer of elements from other sciences, and its practical consequences and (3) The applicability of the model to other fields of science.

INTRODUCTION

Framing Interdisciplinarity

What I am about to do here is by no means the first attempt at the task of framing the concept of interdisciplinarity:

> "Although many have tried to define interdisciplinarity , it still seems "to defy definition". The most widely cited attempts break down interdisciplinarity into components such as multidisciplinarity, pluridisciplinarity, crossdisciplinarity, transdisciplinarity, and even metadisciplinarity. But these subdivisions, it seems to me, throw little light on the theory and practice of interdisciplinarity, in part because they try to grasp points along a fluid, multidimensional continuum. Moreover, because such definitions attempt to confer upon this term a precision it does not possess, they run the risk of missing its essential nature." [Nissani 1995]

This leads Nissani to adopt a minimal definition of interdisciplinarity: "bringing together in some fashion distinctive components of two or more disciplines".

Nissani might be right in stating that this is as precise as a definition can get, if the aim is to define the concepts linguistically in one sentence. I will make no attempts at improving upon this one-liner. Rather, I will make an attempt at clarifying what lies behind the phrasing 'some fashion'. To the just as general and imprecise 'distinctive components', I will not add much clarity on the general level, but I will suggest a method of identifying these components.

The distinction into multi-, trans-, cross- and interdisciplinarity is sometimes used to identify different ways in which some problem might be approached. In multidisciplinarity, a problem is approached from the perspective of several disciplines without integration of the results, such as a multidisciplinary conference with different sciences giving separate contributions to the same subject. Interdisciplinarity (in the context of this distinction – I will use it to cover all instances of science involving more than one discipline) covers the formation of new disciplines drawing upon existing ones. Transdisciplinarity covers instances where one established science makes some contribution to a foreign science. As argued by Nissani [1995] in the above, it is easy to think of hybrids between these distinctions; they form a continuum rather than being discrete categories. Moreover, they are aimed at explaining **why** work across disciplines happen rather than **how**. So while there has been much discussion of why interdisciplinarity is A Good Thing, there still seems to be a need for discussions of how it happens.

Relevance of discussing interdisciplinarity

Interdisciplinarity has played a major role in the development of science and education. Some would say through the last two decades, some would say since the beginning of science. The question is not of importance for my purpose. At least three lines can be identified, by which more recent crossing of scientific boundaries has occurred: The shift towards problem-orientation, a tendency towards increasingly holistic theories and, on a more specific level, the emergence of new technology.

1. Shift towards problem-orientation

In a local context, problem-orientation was established as an ideal in education and research in Denmark in the seventies:

> "During the 70s, two new universities were founded: Roskilde University Centre in 1972 and Aalborg University in 1974. This occurred due to a very strong student movement and, in the case of Aalborg University, an industry that wanted new competence profiles for engineers. Learning by doing and experiential learning were two of the principles that dominated the development of this particular system. Following the student revolts in the late 60s, a strong movement arose in social sciences during the 70s regarding project work as a possible factor contributing to change in society. " [de Graaf & Kolmos 2007]

The ideals outlined above held for the new universities as wholes, not only their engineering departments. The US has seen a similar development [Klein 1990], as has other countries. Working with problems instead of disciplines demands for sciences to integrate, since most (maybe even all) problems have more facets than can be addressed from the viewpoint of one discipline. In other words, instead of having methods and theories determining the problems, the problems determine the theories and methods applied.

2. A general tendency towards a widening of context, increasingly 'holistic' approaches.

The application of various variants of Systems Theory, where objects cannot be studied detached from the system of which they are part is one example. Various social constructivist theories is another: In these theories phenomena such as cognition cannot be studied by looking at the mind alone, since meaning arises socially, between people. There are examples aplenty of theories that not only allow for the consideration of a larger context, but that demand it. Situated learning [Wenger & Lave 1991], in which learning must be studied in a social context, as well as various theories of embodied cognition, following Merleau-Ponty [1962], where the traditional distinction between body and mind is dissolved, are examples within cognitive science. But the reach of the tendency is further. In medicine, holistic approaches are applied to explain phenomena such as psychosomatic illnesses [e.g. Melmed 2001]. Common for all these approaches is that they place a demand on the researcher to take into account phenomena traditionally belonging to other fields.

3. Emergence of new technologies.

The emergence of new technologies, most notably computers, has had deep impact on a number of sciences, such as organisation, psychology and sociology, which all see a change in their area of study, which in turn calls for changes in methods and theory. It has also allowed for the formation of entirely new ones, the most prominent example perhaps being Cognitive Science. Another is the case to be studied here: Human-Centered Informatics.[1]

This above three examples begin to illustrate the diversity of ways in which interdisciplinarity may arise. The first, problem-orientation, is driven by professional and in part societal demands for a revision of method. The shift toward increasingly holistic approaches is driven by a general tendency in academic theory, and the latter is driven by the introduction of an entirely new object of study. It seems obvious that there is a difference in the way interdisciplinarity unfolds in these cases. Nissanis statement that attempts at defining interdisciplinarity as one process is 'conferring upon the term a precision it does not possess' seems plausible.

---

1 It is of great importance to distinguish between HCI (Human-Computer Interaction) and Human-Centered Informatics. As I will explain later, the field of Human-Centered Informatics is often presented as divided into four main areas, one of these being Human-Computer Interaction. By convention, HCI is an acronym for Human-Computer Interaction. So whenever HCI is used in this text, it means Human-Computer Interaction, while Human-Centered Informatics will be spelled out on every occurrence.

Confusion of concepts

I have argued that much of the confusion in framing interdisciplinarity is due to a desire to define it as one uniform process. I have accepted the minimal definition: "bringing together in some fashion distinctive components of two or more disciplines", but this is vague to say the least. The division into multi-, inter-, cross- and transdisciplinarity does not seem to constitute discrete categories, rather they are ends of continua. Much of the confusion as to how interdisciplinarity unfolds between different entities, however they are divided, may be addressed by a division into levels on which import of elements from one science into another happen. For those of us who subscribe to the logic, semiotics and philosophy of C.S. Peirce, such a distinction between levels is readily at hand. This line of thinking also permits for a more formal approach than most previous attempts at framing interdisciplinarity. Approaching interdisciplinarity as logic (although logic, in a Peircean context covers all reasoning), has its obvious drawbacks. Science depends to a high degree on social interaction, and is certainly not carried out only in the cognitive realm of isolated scientists. It also happens in highly complex social networks. This entire side of science is not captured by my approach, since I choose to see science from the point of view of the researcher. The high degree of complexity in interdisciplinarity, that which makes it so difficult to define, demands for such a choice of perspective. I am interested in clarifying how elements move between disciplines. I will do this empirically by looking at the texts. In doing this, I leave the social and interactional elements of science for others to deal with, except for the conclusion of the report, where I will touch on these matters a bit.

Assumptions

To clarify my perspective, I will lay out some of the assumptions underlying this investigation.

1. Research is done by people, and is a cognitive enterprise.
2. Science is production of knowledge.
3. The production of new knowledge depends on existing knowledge.

When looking at science in this way, it is a special case of a classical problem: How does new knowledge arise from existing knowledge? In his famous *The fixation of Belief*, Peirce gives exactly this explanation to why his studies of logic are important:

> "The object of reasoning is to find out, from the consideration of what we already know, something else which we do not know. [EP 111]

When looking at interdisciplinarity, this question becomes yet more specialised: How does new knowledge arise from existing knowledge from more than one discipline?

The fact that I take the point of view of the researcher has the important implication that the transfer of elements is considered as import from other theories. Transfers might of course also be considered export to other disciplines, or interaction between disciplines. For the purpose of my analysis, however, I will consider export as other disciplines importing, and consider only cross-disciplinary import.

What is disciplinarity?

A prerequisite for studying interdisciplinarity is to define disciplinarity. The idea of disciplinarity may be traced back as far as to the middle ages: In part due to external, i.e. professional and governmental demands, research became increasingly specialised during the late middle ages [Klein 1990]. This lead to distinctions between different sciences, the strongest of which is the facultary division into natural sciences, social sciences and humanities.

The distinction between sciences can be opposed from at least two different points of view:

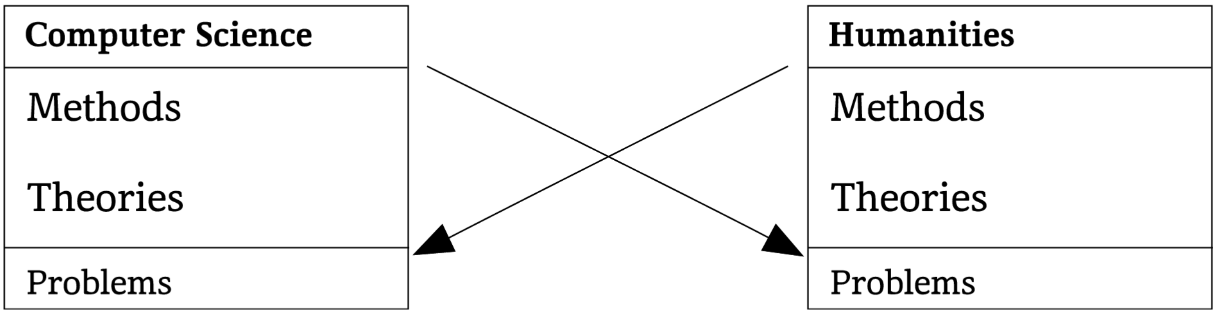
1. The belief in a unity of science, in which scientific divisions are caused by a lack of research, and that more research will enable us to connect all sciences into what would essentially be one science, the science of everything.
2. The notion that scientific divisions are arbitrary, determined by history and tradition, not by natural properties of the sciences themselves.

Both these views contradict the traditional view of distinct sciences existing in and of themselves, almost as if in a vacuum, and they render the concept of interdisciplinarity meaningless. However, whether or not one would subscribe to any of the above, the concepts of both disciplinarity and interdisciplinarity *do* exist. If not elsewhere, then at least in the minds of scientists. Few scientists would deny that research is indeed carried out within disciplines. Tradition, educational backgrounds and social networks as well as conferences and journals devoted to some scientific area keep divisions very much alive. And fortunately so, since it is obvious that no-one is able to oversee every achievement in every science. Yet, most scientists would probably agree to the idea that different scientific fields would have much potential for mutual benefit. One might even chance his arm to say that no science exists independently of others. In any case, an attempt at creating a model of the mechanisms that drive interplay between sciences may be interesting. Choices of seats at conference dinners, casual collegial conversation, peeping into books with interesting covers at the library and chance meetings in hallways undoubtedly play significant roles in that regard. Interesting sociological research may be conducted into this. However, it lies outside the scope of my analysis, in which I will look into the texts and theories, in order to examine the way in which ideas move between scientific fields, in the context that is the discipline of Human-Centered Informatics.

Chapter 1 – The problem of interdisciplinarity

Sciences enrich each other

In order to explain how sciences may interact within the context of Human-Centered Informatics, I will turn to one of the main authorities on the field, Øhrstrøm [2003], who gives the following interpretation of the epistemological nature of Human-Centered Informatics:



"

1. How may one illuminate and enrich problems traditionally studied within the humanities with methods and techniques **brought in** from Computer Science.
2. How may one illuminate and enrich problems traditionally studied within Computer Science with methods and techniques **brought in** from the Humanities"

[Øhrstrøm 2003, my translation and highlights]

It is interesting to note that Øhrstrøm employs the words 'brought in'. This choice of words rests on a conceptual metaphor of methods and techniques as physical entities. In Human-Centered Informatics, scientists from the Humanities may then go to Computer Science, browse its shelves of methods and techniques, and then bring them back to the humanities to use them for illumination and enrichment. In the graphical representation, arrows designate direction and movement.

As such, Øhrstrøms conceptualisation of Human-Centered Informatics as an interdisciplinary science is insightful indeed. However, from my Peircean point of view, I will make the claim that it may be fruitful to consider even more kinds of interdisciplinary transfer. Øhrstrøms model comprises all three universes in a Peircean triadic division: 'theories' are some kind of thirdness, 'methods' are secondness and 'problems' may be seen as firstness. So in Øhrstrøms model, interdisciplinarity in Human-Centered Informatics is transfer of thirdness and secondness to be employed in some way on firstness. And indeed, in many cases this is an accurate description of what happens in interdisciplinary transfer. However, Peirce himself has noted that: “a man needs not the theory of a method in order to apply it as it has been applied already” [CP 7.67]. This suggests that a stronger division between the realm of method, technique or analysis and that of theory may be desirable. When adopting a dynamic version of Peircean semiotics, as I will i my analysis, it becomes even more apparent that interdisciplinarity may unfold in more manners than those lying within the scope of Øhrstrøms interpretation.

Preliminary work

As mentioned in the introduction to this thesis, I have done some preliminary analyses on the subject of interaction between Cognitive Science and HCI

[Hove 2007]. By using a distinction adopted from Billman [1999] into different so-called *realms* of representation, those being external, mental, computational, theoretical and physiological realms of representation, I identify some forms of interplay between HCI and Cognitive Science. Further to, I conclude that the division of the sciences into realms is not in itself interesting. What is interesting is the dynamics and movements between realms.

The work with the division into realms was one of the main inspirations to taking a peircean approach to the problem. Although the distinctions into realms employed in Hove [2007] and those employed here are different, they share at least the characteristic that they are categorisations of scientific phenomena, and it would be reasonable to assume that the idea of movements between realms would be interesting to look at when studying interdisciplinarity.

The movement metaphor

It would be uncontroversial to most to accept the premise that new knowledge does not appear out of nothing. It must be based on existing knowledge, new knowledge, sensory information etc., the nature of which I do not wish to go into. In interdisciplinary science, new knowledge arises out of existing knowledge from different scientific disciplines. Øhrstrøms notion that this happens by applying methods, techniques and theories from one science (Computer Science) to another (Humanities) is very insightful. However, any elaboration of how this might happen falls outside the scope of Øhrstrøms analysis. He makes no assumption that the theories, methods and techniques from Computer Science lend themselves to direct application to the Humanities. On the contrary, he argues that considerable energy is required in order to develop a scientific self-understanding in the highly interdisciplinary field of Human-Centered Informatics. The tension-field of Computer Science and the Humanities is interesting and fruitful, as years of research within the field has proven.

Lots of obvious problems arise when bringing together such diverse fields. First of all, there are obvious differences in the domains that the sciences are concerned with. Grossly oversimplified, the domain of Computer Science is computers and the domain of the humanities is humans. It seems obvious that these domains are very different in nature, and it seems improbable that the same methods and techniques apply to both domains without modifications of some kind. Secondly, these domains belong to different scientific traditions altogether. These different traditions traditionally employ very different methodologies in analysing their domain. In the simplest distinction, attributed to Dilthey, the natural sciences seek explanation, whereas the humanities seek understanding. [Mautner 2000, p. 144]. This view is deeply embedded in the self-understanding of the different sciences, and consequently, the traditional methods taught at the different faculties reflect this view. It might be said that the humanities use interpretative methods of explanation, whereas the natural sciences use causal models of explanation. [Faye 2000, p. 56]. With these seemingly profound differences in mind, it is difficult to imagine elements from one of the sciences being 'moved' directly into the other. On the other hand, this sharp distinction between the humanities and the natural sciences is quite naïve.

There are certainly similarities and overlap in the domains. Human-Centered Informatics is a prime example of a field that acknowledges this: In lies in the name of the discipline that the domain of Human-Computer Interaction crosses the traditional scientific boundaries. Artificial Intelligence where computers attempt to simulate the human brain is another example, as well as more modest (although no easier) tasks such as automatic translation, where hermeneutics become part of the domain of Computer Science.

The differences between the scientific traditions are certainly real. In an interesting and provocative account, however, Feye [2000] argues for a methodological unity of science, on the grounds that the cognitive means and goals are the same in the two sciences, and that the methods are essentially the same as well. Føllesdal et al. [1990, pp. 119nn] argue that the hypothetico-deductive method, while traditionally belonging to the natural sciences, is applied in all sciences including the humanities, social sciences and ethics. These views are quite radical, but even within hermeneutics is has been granted that while humanities should not imitate the natural sciences in all respect, the general ideals from the natural sciences, such as objectivity and the aim for generalisation, are not rejected [Paahus 1995]. Finally, we have Øhrstrøms claim that Computer Science and the Humanities can benefit from the theories and methods of one another.

To sum up, there is certainly possibility for 'moving' elements, even between sciences traditionally considered profoundly different. However, it is not simple to account for these movements, since they may happen at different levels. I will argue that they happen on three: That of the domain, that of analysis, and that of theory.

In addition, when 'moving' problems, methods and theories between science, they might somehow change in order to fit the sciences into which they are imported:

> "Now although a man needs not the theory of a method in order to apply it as it has been applied already, yet in order to adapt to his

> own science the method of another with which he is less familiar, and to properly modify it so as to suit it to its new use, an acquaintance with the principles upon which it depends will be of the greatest benefit. For that sort of work a man needs to be more than a mere specialist; he needs such a general training of his mind, and such knowledge as shall show him how to make his powers most effective in a new direction. That knowledge is logic." [CP7.67]

The above citing from C.S Peirce is one attempt at explaining how movements between sciences happen. The model I will present later is based on Peirces philosophy, and is an expansion of the above view. What he argues is that methods may be imported to a science without necessarily importing the theory of that science. On the other hand, the method should be modified, a modification that is somehow connected with 'the principle on which it depends'. In order to make clear how this is done, Peirce suggests his logic. I will accept that suggestion and employ Peircean logic along with other elements of his philosophy in trying to explain how movements between sciences happen.

Chapter 2 - Conceptualisation of movements

Introduction

I have argued that interdisciplinarity happens through movements. In this chapter, I will conceptualise these movements in a way such that they may be used for analysis. I will do this by presenting a number of tools that may be useful in describing interdisciplinarity. I will describe the tools, not as much as they are presented by their original authors, but in the way I intend to use them. None of the tools are designed for the analysis of interdisciplinarity as such, but I will argue that this particular combination is. In some cases, only part of a theory is useful for this study of interdisciplinarity. In these cases, I have reduced them in complexity. Other theories need a bit of adjustment in order to be applicable. In the case of Richmonds theory on |>*, this has added complexity to the theory. My model of analysis rests on 4 main components:

Semiotics is employed as a way of establishing the units of analysis.

Universes of experience are prerequisite for understanding |>*, and they establish a view of science that allows for movements.

|>* is used to describe movements in science.

Similarity is a way of seeing connections between sciences, taking the idea of movements from intra- to interdisciplinary.

The elements underlying the model might be graphically represented as:

<table>
<tr><td colspan="2">THE MODEL</td></tr>
<tr><td>|>*</td><td rowspan="2">The notion of similarity</td></tr>
<tr><td>Peircean Universes of experience</td></tr>
<tr><td colspan="2">Peircean semiotics</td></tr>
</table>

This is to be perceived as a layer-cake, in which the underlying elements are prerequisite for the ones on top. The peircean idea of universes of experience is part of his semiotics. Richmonds |>* is a reconceptualisation of this idea as well as ideas from logic. Since similarity plays a prominent role in the model of interdisciplinary movements, I will also present a notion of similarity, which is based loosely on Peircean ideas, although the notion of similarity is a much simplified one, since it is part of an extensive discussion for which this report does not leave room. In the following, I will outline the four components underlying the model as well as their functions:

Semiotics

Sowa [2000] introduces the metaphor *Knowledge Soup* to characterise bodies of knowledge. The soup may contain coherent chunks, such as theories, but seen as a whole the soup is “fluid, heterogeneous, ever changing and often inconsistent in nature” [p. 348]. The knowledge in the domain of science is such a soup. When looked at interdisciplinarity, the body of knowledge comprising multiple sciences is an even larger bowl of soup with even more complexity, inconsistency etc. For the purpose of analysis, as well as for that of conducting interdisciplinary science, this presents obvious problems. One of these problems is to establish the units of analysis. Another is simply to derive meaning of vast bodies of knowledge. One solution suggested by Sowa is semiotics. “The knowledge soup consists of collections of signs. Images, symbols, words and concepts with associated feelings” [p. 394]. The units of analysis, then, is signs. Sowa mentions images, symbols, words and concepts. These things on paper are then associated with something, and through this association they are signs with meaning.

The semiotics of C.S Peirce

Several reasons may be given for my choice of the semiotics and philosophy of CS Peirce. First of all, it is somewhat dictated by the choice of |>* as a framework for analysis. But, perhaps more importantly, it is a choice. I have found Peircean semiotics useful in describing many aspects of the world, especially when these descriptions are of a cognitive nature. The general idea and entire line of thinking presented here is thus conceived from and inspired by Peircean philosophy.

“if I am asked to what the wonderful success of modern science is due, I shall suggest that to gain the secret of that, it is necessary to consider science as living, and therefore not as knowledge already acquired but as the concrete life of the men who are working to find out the truth” [CP 7.50]. Science is done by humans, and as such, it is a cognitive enterprise. This is another reason for employing peircean semiotics: It is a theory of cognition. A sign is “something by knowing which we know something more” [CP 8.332], and this sign-process itself is a process of cognition; Pierces' theory of signs is so closely related to his theories of logic (in many of Peirces writings, logic is more or less equivalent to cognition) that it is often difficult to distinguish between the two. This aspect of Peircean semiotics and logic is exactly what Richmond goes into in his contemporary interpretation of Peircean semiotics/logic, |>*, which I will present a bit later.

Conceptualisation of science

In the following, I will bring the context of scientific enquiry into the triadic terminology of Peirce. This will be done by outlining the most general triad, that of first-, second- and thirdness, and then specifying it into the view of science that forms the basis of my model.

Universes of experience

Firstness/Secondness/Thirdness

“My view is that there are three modes of being. I hold that we can directly observe them in elements of whatever is at any time before the mind in any way. They are the being of positive qualitative possibility, the being of actual fact, and the being of law that will govern facts in the future.” [CP 1.23]

The notion of three modes of being is as general as theory gets; it comprises 'elements of whatever is at any time before the mind in any way'. As such, this notion may be applied to anything. On the other hand, it is too general to make use of in itself, without some explanation to how it is applied. Before doing that, however, I will give an overview of the three categories as they are described by Peirce himself:

Positive qualitative possibility, or firstness, is described by Peirce as:

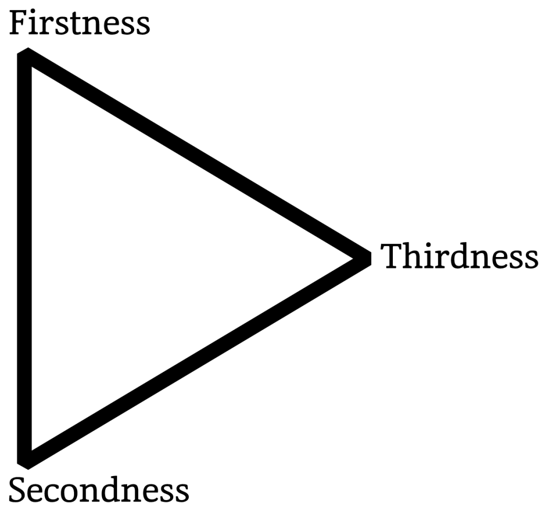
> “Firstness is the mode of being which consists in its subject's being positively such as it is regardless of aught else. That can only be a possibility. For as long as things do not act upon one another there is no sense or meaning in saying that they have any being, unless it be that they are such in themselves that they may perhaps come into relation with others. (...) We naturally attribute Firstness to outward objects, that is we suppose they have capacities in themselves which may or may not be already actualized, which may or may not ever be actualized, although we can know nothing of such possibilities [except] so far as they are actualized.” [CP1.25]

Actual facts, or secondness:

> "If I ask you what the actuality of an event consists in, you will tell me that it consists in its happening then and there. The specifications then and there involve all its relations to other existents. The actuality of the event seems to lie in its relations to the universe of existents. (...) Actuality is something brute. There is no reason in it. I instance putting your shoulder against a door and trying to force it open against an unseen, silent, and unknown resistance. We have a two-sided consciousness of effort and resistance, which seems to me to come tolerably near to a pure sense of actuality. On the whole, I think we have here a mode of being of one thing which consists in how a second object is. I call that Secondness." [CP 1.23]

And thirdness, law governing the future:

> "Five minutes of our waking life will hardly pass without our making some kind of prediction; and in the majority of cases these predictions are fulfilled in the event. Yet a prediction is essentially of a general nature, and cannot ever be completely fulfilled. To say that a prediction has a decided tendency to be fulfilled, is to say that the future events are in a measure really governed by a law. (...) A rule to which future events have a tendency to conform is ipso facto an important thing, an important element in the happening of those events. This mode of being which consists, mind my word if you please, the mode of being which consists in the fact that future facts of Secondness will take on a determinate general character, I call a Thirdness." [CP 1.26]



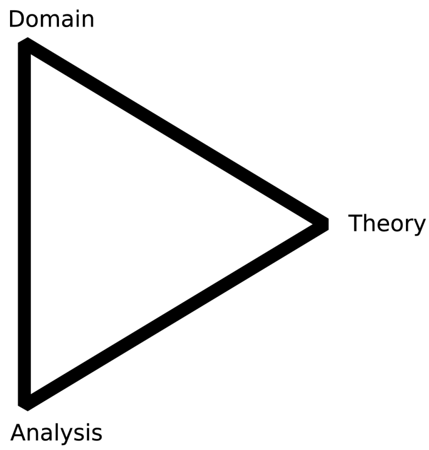
The graphical, triadic illustrations often employed when presenting peircean semiotics have never actually been presented by Peirce himself. They may be considered a later, pedagogical contribution, and as such, I will take the liberty of presenting the triadic way of thought somewhat differently, namely in the manner of [Richmond 2005], as I have done it above. The difference between Richmonds representation and the more traditional one, is that Richmond has chosen to rotate the triangles by 90°. This is actually done in order to increase the pedagogical quality of the representation;

> "Firstness: Ideas (in the platonic sense), "airy nothings", mere possibility of being actualized, so, "floating" [at the top].Thirdness: Habits, or, more exactly, habit-taking, tending to bring firstness and secondness into relationship in futuro, so to the right [in futuro, what happens when pressing a play-button, which the triad resembles] Secondness: Brute events, actions and reactions, existential and earthbound, so, "sinking" to the bottom of the diagram." [Richmond 2005]

Peirces own definitions, as reproduced above, I will adopt as the most general triad conceivable, that of 'universes of experience'. (firstness/secondness/thirdness). While in the above, I have let the words of Peirce speak for themselves, I will interpret these definitions in the following, by means of a specification. This specification takes the form of a gradual application, which serves as an explanation to how this peircean conception of experience may be applied to an analysis that takes the viewpoint of Human-Centered Informatics. In this way, I take experience to be the most general universe, in that its domain is every potentiality conceivable (or yet to become conceivable). Universes of experience subsumes universes of science, in which firstness is specialized to comprise only those possibilities that come into effect through a special type of 'brute force', that of science.

Universes of science

The move from universes of experience to universes of science is has the character of an application, where science is perceived as subsumed by experience, i.e. a more specific form of experience. On this line of thinking, scientific processes may happen in, and knowledge may reside in three different scientific universes: That of the Domain, that of Analysis and that of Theory:



Domain

> "the first [universe] comprises all mere Ideas, those airy nothings to which the mind of poet, pure mathematician, or another might give local habitation and a name within that mind. Their very airy-nothingness, the fact that their Being consists in mere capability of getting thought, not in anybody's Actually thinking them, saves their Reality." [CP 6.455]

In an earlier citing, this was dubbed 'qualitative possibility'; in science, this is what may be called 'the world', that which is somehow observed or analysed. Yet, it is nothing in itself, because no analysis may contain every aspect of this "world". When I take the concept of 'domain' to be subsumed by possibility, it is therefore not to be taken as one particular domain, or as everything comprising the domain in question. Rather, it may be said that the domain is **made up** of such endless possibilities, only some of which may come into effect, or, perhaps more precisely, the domain has its very existence because of the existence of some category of possibilities;

> "If you ask what mode of being is supposed to belong to an idea that is in no mind, the reply will come that undoubtedly the idea must be embodied (or ensouled -- it is all one) in order to attain complete being, and that if, at any moment, it should happen that an idea -- say that of physical decency -- was quite unconceived by any living being, then its mode of being (supposing that it was not altogether dead) would consist precisely in this, namely, that it was about to receive embodiment (or ensoulment) and to work in the world. This would be a mere potential being, a being in futuro; but it would not be the utter nothingness which would befall matter (or spirit) if it were to be deprived of the governance of ideas, and thus were to have no regularity in its action, so that throughout no fraction of a second could it steadily act in any general way. For matter would thus not only not actually exist, but it would not have even a potential existence, since potentiality is an affair of ideas. It would be just downright Nothing." [CP 1.218]

A similar thought can be found in Kuhn (1968):

> "Perhaps immediate experience should be set aside as fluid, and we should discuss instead the concrete operations and measurements that the scientist performs in his laboratory.(...)Science does not deal in all **possible** laboratory manipulations. Instead, it selects those relevant to the juxtaposition of a paradigm with the immediate experience" p. 125nn

Although possibly not meant by him as such, what Kuhn hints at above may be perceived as an application of the peircean idea of firstness to science in general. Common to the ideas of Peirce and Kuhn in the above citings (though I am not implying any further commonalities between the two) is the potentiality of firstness. 'Fluid experience', or potential existence of ideas is what constitutes what I have labelled 'the domain'. The domain is thus not 'everything that exists', nor is it 'nothing in itself, emerging only as experiences,' as in solipsist ways of thinking (solipsism, then, actually being the 'deprivation of of the governance of ideas'). A domain is real, in so far as it carries a potential of becoming actualised, embodied, conceived. In the words of Peirce “they are such in themselves that they may perhaps come into relation with others”. This 'relation with others' is a triadic one. By means of some thirdness, into which will be looked later in this text, these firstnesses become actualised in the form of a secondness. In the context of science, I will use the concept 'analysis' as a concept for describing this specialized form of secondness.

Analysis

> “The second Universe is that of the Brute Actuality of things and facts. I am confident that their Being consists in reactions against Brute forces, notwithstanding objections redoubtable until they are closely and fairly examined.” [CP 6.455]

So, the universe of analysis covers instances of 'brute force' and actualities. This immediate experience, or brute force, is the method, or practice, of science. I will call that 'analysis', referring to the product 'an analysis', i.e. something actual. although not necessarily on paper. The analysis might be only inside the head of a scientist, but it is to be considered an actuality regardless. It also refers to the act of analysing. Analysis should be taken to cover all sorts of practical work in science. As such, in some instances the word 'practice' is more appropriate, and is sometimes used in my analysis.

Theory

> "The third Universe comprises everything whose being consists in active power to establish connections between different objects, especially between objects in different Universes. Such is everything which is essentially a Sign -- not the mere body of the Sign, which is not essentially such, but, so to speak, the Sign's Soul, which has its Being in its power of serving as intermediary between its Object and a Mind." [CP 6.455]

Because the domain consists of nothing but mere possibilities, some mediation is necessary in order to look at and comprehend it. The scientist will use some theory to arrive at actualities from the endless possibilities. That the concept of a theory is perceived as 'A law governing the future' entails some sort of process, a process impacting the relation between 'domain' and 'analysis';

> "every idea has in some measure, in the same sense that those are supposed to have it in unlimited measure, the power to work out physical and psychical results. They have life, generative life." [CP 1.219]

This view of ideas, or theories, as the driving forces of processes (entailing some kind of temporality), corresponds with the notion of ideas, i.e. Domains, as something that **is** by virtue of **its being *in futuro***. In other words, a process, driven by some idea, is required to realise these possibilities. That is the 'generative life' of theory. In order to be valuable, theories make, to use the words of Peirce, 'some kind of prediction'.

|>*

Outline of |>*

When looking at how knowledge 'moves' between sciences, the interesting thing is not as much the division into domain, investigation and theory. This is merely the outer theoretical framework for analysis. Of particular interest here are the dynamics and the movements between sciences, and the way in which they take place in the universes of science. These dynamics are hinted at by Peirce himself, and should be seen as inherent in peircean ways of perceiving reasoning. The dynamics of triads have, however, been more clearly laid out by the contemporary peircean scholar Gary Richmond. I will employ his way of thinking about movements in triads when investigating into the nature of interdisciplinarity.

> “The thematization and exploration of the six vectors (especially as they interpenetrate in hierarchies of constraint, dependence/independence, etc.) aims at bringing new light, and ultimately a more systematic treatment, to some of the difficult issues which arise especially in semeiotic analysis. In short, the vector issue invites treatments involving graphically logical aspects of dependence and constraint, correlation, as well as the “living” reflection of the categories by the semeiotic triad.“ [Richmond 2005]

Gary Richmond suggests a way in which movements within these triads take on the form of 'vectors'[2]. While 1ns, 2ns and 3ns:

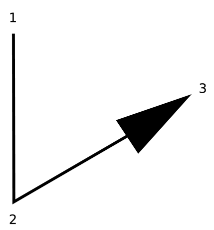
1ns

|>3ns

2ns

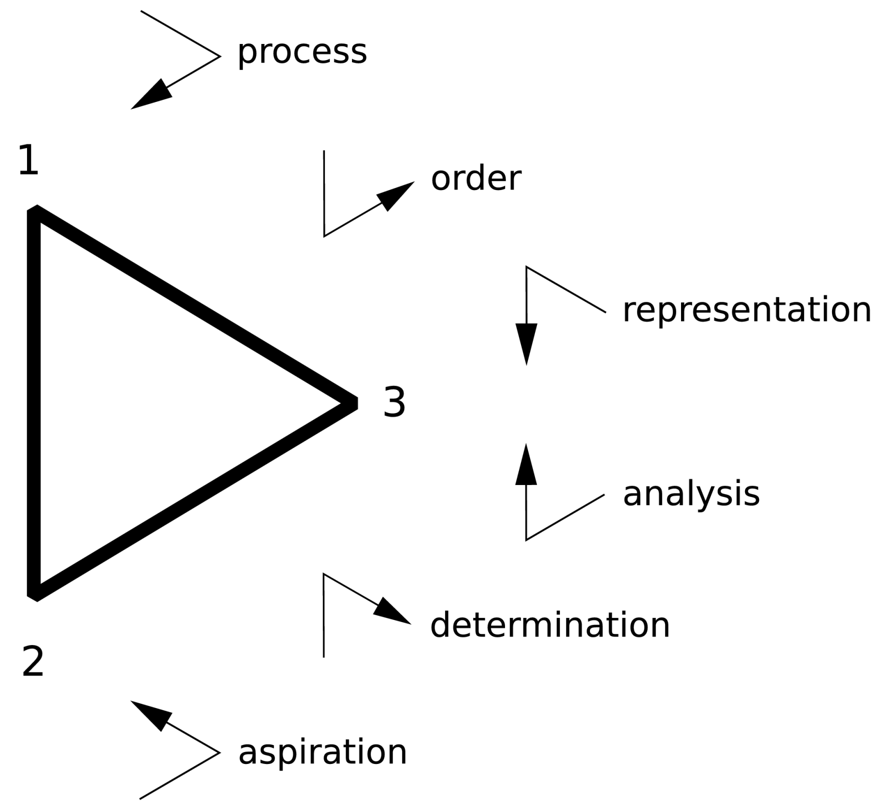
should be perceived as “occurring all-at-once, each necessitating each other, no one more fundamental than the others” [Richmond 2005], what Richmond introduces into this way of thinking is “vectorial movements” in the trichotomic relationship. The trichotomies become dynamic (or, the dynamics inherent in the trichotomies become explicated and even visualised), by focusing on processes between 1ns, 2ns and 3ns. These processes, are then represented as diagrams showing six different possible processes, or vectorial movements (which is every possible vector).

Vectors can be represented graphically, or they can be written in text:



Is equivalent to writing 1/2/3>

2 Richmond has pointed out, that the notion of the vector has no connection, other than its graphical appearance, with the mathematical concept, and should not be confused with this. [Presentation at ICCS 2006]



Vectorial analysis is applied at two different levels:

1. Internal vectorial movements inside trichotomic relationships. This is applied to a clarification of different relations between domain, analysis and theory within theoretical development and scientific practice.
2. The concept of “the chiral cycle”. By taking as its starting point the idea of vectorial movements, it may be analysed how different movements take place **between different trichotomies.** This is applied to an analysis of how different theories may exist in interplay, which may implicate all elements under the meta-trichotomy “domain-analysis-theory”.

Initiation/Procœss/Product>

Even through Richmond suggests that |>* is useful in linguistic analysis, his main application is to collaborative work. In this context, the diagrammatic representations provided by |>* may be used for joint diagram manipulation (diagram manipulation is a key element in reasoning in some of Peirces writings). Since my aim is different from this, so is my conceptualisation of |>*. I have adjusted it for analysis of scientific processes, as well as for interdisciplinarity, by breaking the idea of the vector into three elements: Initiation, Process and Product.

It is obvious that every vector goes through three universes in the triad. As should be clear, the three universes of a triad are closely interconnected. The elements, or 'corners' of the vectors themselves, however, may be said to have certain properties of their own. Like triads may be decomposed into firstness, secondness and thirdness (although this distinction, as should be clear, is only a means of decomposing the triads into units of analysis), vectors may be decomposed into three elements; a beginning, a middle and an end. They illustrate movements or processes, and may be dubbed:

initiation/process/product>

All these elements of a vectorial movement, initiation, process and product may happen in all universes, as the six vectors illustrate. Richmond orders his vectors in pairs, in a way such that he puts emphasis on the process part of the vector; vectors that have their process happening in firstness (2/1/3 –

determination and 3/1/2 – representation) are grouped, processes in secondness have 1/2/3, order and 3/2/1, analysis. In thirdness 1/3/2 evolution and 2/3/1 aspiration. Apart from placing emphasis on the process element, the two vectors of a pair also represent each others opposites, a fact that may be the reason for Richmonds choice of groupings. Depending on the purpose of ones analysis, the vectors might of course also be ordered by their initiation or product.

Initiation

Vectorial operations between sciences are initiated by some kind of similarity. In the philosophy of Pierce as well as in other cognitive theories similarity or resemblance are given great importance in the initiation of processes of reasoning. I will dwell a bit more with this later.

Initiation by Domain and Analysis is roughly the same process. It happens due to some kind of similarity.

Initiation by theory. The category of inter-theoretical vectorial operations which are initiated by theory differs from the others. Theory import when it happens in the theoretical universe, is initiated by the theory itself. As such, the choice of theory does not rest upon any kind of likeness either in domain or analysis. It rests on a somewhat different kind of similarity, because it need not be due to likeness in intrinsic properties of the theory. The reason for similarity may also be of a social nature, like theoretical movements in related sciences or more general theoretical tendencies and scientific fashion. The choice of theory might also be of a more arbitrary nature, which in this context should not be always be taken in its usual derogatory form. Rather, this form of initiation contains a number of powerful scientific instruments, such as intuition, inspiration from other scientists and general gut feelings that some theory might be beneficiary to ones work. Thirdness has been said to have 'generative life', and it is this generative life that is realised through initiation by theory; new connections are made between domain and analysis that would not have been possible through the use of other theories.

Process

This is where the actual import happens from one science into another.

Process in the domain involves somehow making problems, or elements of the foreign domain, part of ones own. This may happen out of a need to expand ones own domain to incorporate elements traditionally studied within another. It may also be of a more epistemological nature, i.e. to consider problems of another science problems of ones own.

Process in analysis is import of methods from another domain. In an earlier citing Peirce stated that “a man needs not the theory of a method in order to apply it as it has been applied already”. This is the general idea of process in analysis, import of a practical method. a way of doing things, from another science, but without theoretical import. It often has the character of 'trying something out'

Process in theory is application of a foreign theory in ones own science. In some cases, such an application may take on the form of deduction, deriving the consequences of using a different theory.

Product

The product of a vector takes on very different forms depending on whether it is product in the domain, analysis or theory.

Products in the domain

Thinking about 'products' in the context of secondness and thirdness is comparatively straightforward. Processes come into effect in those universes by some kind of force and by habit-taking, respectively. However, a product belonging in firstness, the world of mere possibilities, airy nothingness and things that only exist because of their potential of coming into being, are somehow harder to envisage. Firstness has sometimes (and also by Richmond) been labelled 'ideas'. This word may be the most helpful in understanding firstness-products. They somehow allow for a cycle of reasoning to move on from firstness. In one of his examples, Richmond [2006] uses the vector of aspiration (which is one of the vectors that 'produce firstness') for the following interpretation:

..facilitating the work towards the accomplishment of its goals.

2/3/1> ..takes on new habits of collaboratorial conduct deemed reasonable...

A virtual community (or project group) after due critical consideration...

This vector is taken out of an attempt on Richmonds part to apply |>* to collaboratory and virtual community development, something quite unrelated to the analysis in this report. This particular example has been chosen because of the word 'facilitate'. In the above, the process is in thirdness, so it is habit-taking. The habit-taking itself, however, is not the product, because the product is in firstness. The product is 'facilitation', i.e. the production of the facility or possibility for doing something else, without actually doing it. So products in firstness are vague and maybes, but they are products never the less.

Products in analysis

When the product is in secondness, it is some kind of force, or realisation of something; it is tangible in some way. These products may be what the name of the universe suggests: analyses. It may also be some kind of action or practice – brute force. Tangibility here is to be taken in a very broad sense: In some cases, thoughts may even qualify as secondness.

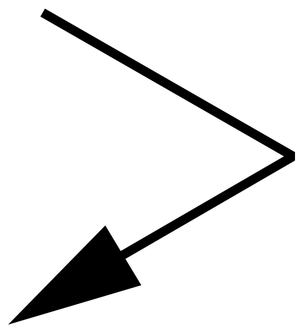
Products in theory

Products in theory are easily accounted for: They are habits and theories.

The vectors

Here, I will present the six different vectors. In cases where Richmond has given examples of the vector, I will cite these. I will the interpret the vectors as initiation/process/product>, in order to put into words the main inference of the vector in the context of interdisciplinarity.

Process 1/3/2>



As examples of this vector, Richmond [2006] provides:

'evolution':

Chance variations occurring...

1/3/2> ...in a structural context allowing for new habit-taking...

...resulting in a evolutionary structural change.

And the analogous 'inquiry':

Retroduction of hypothesis (abduction)...

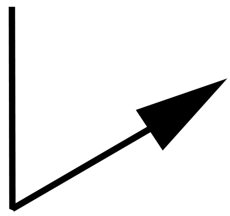
1/3/2> ...proceeds to its implication for testing...

...leading to the construction of experiments and the actual testing of the hypothesis (induction)

Initiation by some variation (or likeness) in the domain is a classic form of science. As an example of enquiry taking the form of this vector is the typical example of some observation in a domain that somehow 'misfits' the knowledge previously obtained, when considered in some theoretical context. What is important to note with regards to this particular vector is that it is 'application of domain to theory' rather that the application of theory to a domain, i.e. 'selecting theories that fit observations made in the domain.'

In the context of interdisciplinarity, the main inference behind this vector is: Sciences agree in problems posed by their domain, so they are likely to benefit from the same theories in analysing them; If we encounter problems in our domain that our theories cannot address, we have to look at the theories that other sciences use to address those problems.

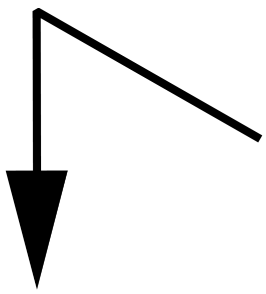
Order 1/2/3>



Similarly to the process vector, the vector of order is initiated by likeness of domains. The process, and the import, happens in the analytical universe, so this is import of method from another science. This is import of methods and analytical practices, rather than of theories, so the main inference is:

Sciences agree in problems, so it may be possible to apply similar analytical practices to develop theories about the domain.

Representation 3/1/2>



...creates a complex symbol...

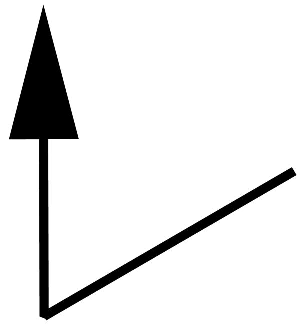
3/1/2>An 'interpreter' of some field...

...representative of a universe of his understanding.

Like the vector of analysis, the representation-vector is initiated by theory. It is the application of a foreign theory that somehow seems interesting, often due to some kind of likeness to the theories of ones own domain. Unlike what happens in the analysis-vector, however, the methods and practices of the theory are not applied directly. Rather, representation is an assumption that ones domain present similar problems and display similar characteristics to the one from which the import (in firstness) happens. These are derived through trying them in practice.

The main inference: There is some likeness between theories of two fields, so it may be beneficial to assume a likeness in problems, and try putting them into practice. This is one special way of developing hypotheses, applying a theory to new problems, resulting in a method that lends itself to testing.

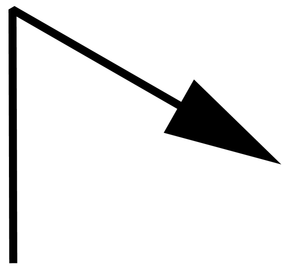
Analysis 3/2/1>



As the name of the vector suggests, this is re-analysis of some domain, on the basis of some theory. The methods and practices of a new or modified theory is applied to the domain. This might be called direct application, since it is conceptualisation of ones domain through a different theory.

The main inference: There is some theoretical community between scientific fields, so it is likely that the practice or methods of the foreign science may be helpful in conceptualising ones own domain.

Determination 2/1/3>



...a sign

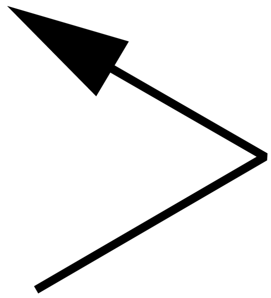
2/1/3> ...for an interpretant.

The object determines...

The vector of determination has its product in theory. This is a way of habit-taking that evolves from secondness, or practice. It is empirical work initiated by observation, leading to a process in the domain; through this process, that which is the product is some kind of revision of the theory, or even a new theory altogether, depending on other processes involved in the system of reasoning.

The main inference: Sciences agree in analytical practice, so they are likely to also share epistemological characteristics (and pose similar problems), to be taken into account when developing theory.

Aspiration 2/3/1>



...facilitating work towards the accomplishment of its goals.

2/3/1> ...takes on new habits of collaboratorial conduct deemed reasonable...

A project group after due critical consideration...

Like determination, aspiration is initiated in secondness, and as such, is based on some kind of practice or actions. As seen from Richmonds example above, it is taking on of new habits. In interdisciplinary work, the 'habits', or theories that are taken on come from other sciences. Such theoretical import may facilitate new ways of conceptualising the domain.

The main inference: Sciences agree in practice, so they are likely to benefit from the same theories in their conceptualisation of the domain.

Forms of reasoning

Richmond explains the workings of |>* mostly by example. Another one of Richmonds examples is his re-interpretation of the forms of reasoning, in which he uses Peirces own well-known bean-in bag example. Instead, however, of presenting them in the traditional way, of syllogisms, he applied |>* to them:

> “
>
> Rule/case/result:
> result (these beans are white)
> |>rule (all the beans from this bag are white)
> case (these beans are from this bag)
>
> abduction (representation vector):
> this handful of beans that I find on the table are white;
> 3/1/2 |>All the beans from this particular bag are white,
> this handful of beans are possibly from this bag.
>
> deduction (analytical vector):
> these beans will necessarily be white.
> 3/2/1|> All of the beans from this bag are white,
> these beans are drawn from this bag;
>
> induction (determination vector):
> all these beans are white;

2/1/3 |> all the beans in the bag are probably white.

These beans are drawn from this bag,

" [Richmond 2005]

This is an example of how different vectors produce different kinds of knowledge through different processes. Induction is initiated by some observation, 'case', or in practice, and result in some rule. This clearly illustrates how induction represents movements from the specific to the general. Abduction and deduction are both initiated in thirdness, by a 'rule'. However, their processes happen in different universes, making the nature of reasoning different. Abduction is 'trying something' or making a guess: The product is in secondness or 'case', so the knowledge produces applies only to the particular case. Deduction, on the other hand, has its product in firstness. Such results, as mentioned earlier, are only possibilities, but the results are strong nevertheless because of their general nature. Abduction is often perceived as being based on hypothesis, whereas deduction is based on a theory. In the above interpretation, however, they are both initiated in thirdness. This shows that in essence, there is no difference between a theory and a hypothesis: They are both a 'habit' or a 'rule'. The difference lies in the way in which this habit is applied. Also, these vectors of reasoning seldom stand alone. More often, they are parts of more intricate systems of reasoning. Since abduction, for instance, gives products in secondness of an uncertain nature, it is of great importance to also consider the vectors preceding and following in systems of reasoning. Exactly this is the idea behind vectorial cycles.

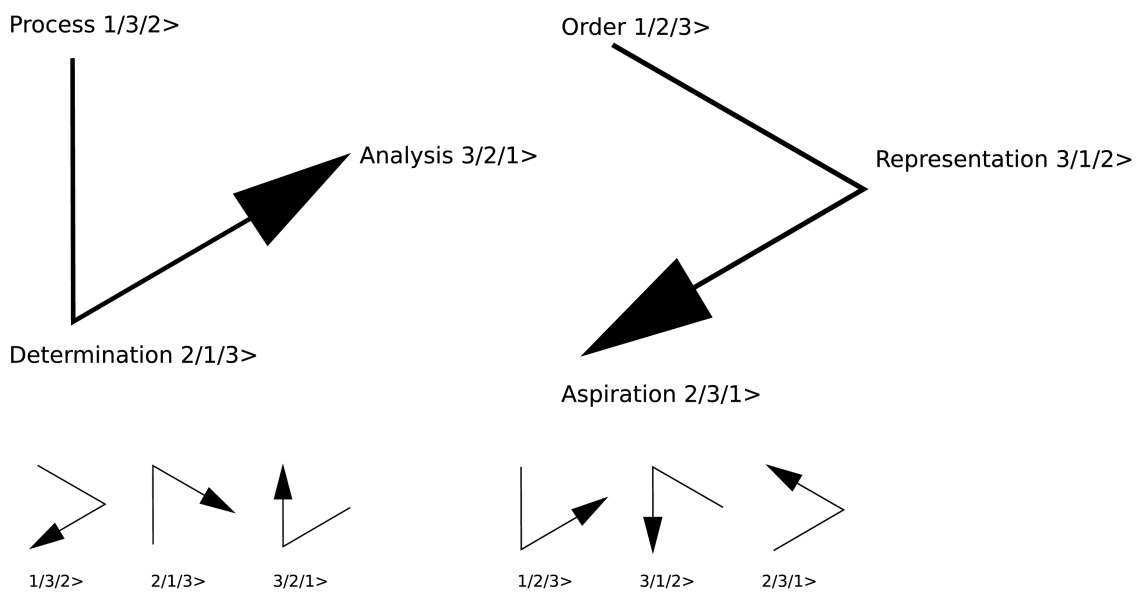
Cycles

Vectorial operations do not happen in isolation. They may also be seen as parts of larger systems of reasoning, in what Richmond (2006) calls 'vectorial cycles'. He takes an interest particularly in what he labels 'the chiral cycle' [Richmond 2006]:

> "Besides being arranged as three pairs of opposites, the six vectors can themselves be related trikonically to suggest possible relations of inter-penetrations of vectorial-categorial movement through rather many dimensions of triadic (and higher-adic) relationships. The most important of these is the *chiral cycle* involving all six vector patterns in a "melding order" which may prove to be the most promising for the proposed analysis-synthesis being considered (...) Here, in moving from one vectorial pattern to the next, the last category of a given vector becomes the first of the following, so this can be seen as a kind *melding* of the six vectors continuously one into the other. The last of the six itself melds into the first so that the pattern may be repeated" [Richmond 2005, p. 166]

This circular way of perceiving vectorial movements blends in nicely with my interpretation of vectors as initiation/process/product>. However, as will be seen, the cycle identified by Richmond, is limited to initiation by firstness. Richmond makes a division of the chiral cycle into two patterns: Right-skewed (clockwise) and Left-skewed (counter-clockwise):

[Richmond 2006, p.167, sequential illustration added by me]



The diagram above clearly shows how vectors tie together in a semiosis-like manner. The overarching vector in the counter-clockwise pattern, 1/2/3>, the vector of order, seen on the left break down into the vectors 1/3/2>, a move from firstness to secondness by means of thirdness, leading to 2/1/3>, a move from secondness to thirdness, and finally 3/2/1>, closing the cycle. These cycles are re-represented below the vectors in order to clearly show that a vector, in its own right, may be perceived as a sequence of three vectors. These, in turn, are themselves sequences of vectors, and so forth. In my analysis, however, I will keep at the level illustrated above, i.e. I will look at one vector at a time in the form of three vectors in sequence. Apart from the two vectors in the chiral cycle, however, I will look at others:

The overarching vector in the left-skewed pattern is process, which corresponds to the peircean perception of scientific inquiry:

> "In Peircean inquiry at least, the following three steps occur: (1) an abduction – in experimental science, retroduction to an hypothesis –

gives rise to (2) the deduction of what would necessarily follow if the hypothesis were correct, and upon which an experiment may be constructed, followed by (3) the actual inductive testing of the hypothesis. (Such inquiry expresses, nearly to perfection, the process vector).

abduction (the case is possible)

1/3/2|> deduction (the result is necessary)

induction (the rule is probable)

" [Richmond 2005, 10]

The chiral cycle is interesting because it demonstrates how chains of reasoning may become full circle, that is, if a problem of some kind is noted in one universe (1ns, 2ns or 3ns), the chiral cycle may clarify the line of thought involved in going through the cycle in order to eventually address the problem itself.

However, my analyses of academic literature suggest that movements need not unfold in these exact manners, and I perceive Richmonds presentation of these as a kind of idealised knowledge production. The particular cycles pointed out by Richmond are always initiated in firstness, the domain. However, Richmond does not argue that his exact cycle is the only one conceivable (pure mathematics tell us that it is not), only that is holds a special status.

His argument for giving the chiral cycle a special status in his 2006 paper is based upon citings from Peirce himself, and whether or not this assertion is true when vectorial analysis is applied to interdisciplinarity as is the case here, is a question best left for the analysis to answer.

This is especially interesting in connection with cross-disciplinary reasoning, since they 'suggests possible relations of inter-penetrations of vectorial-categorial movement through **rather many dimensions of triadic (and higher-adic) relationships**'. In order to keep complexity at a reasonable level, I will keep a somewhat forced focus on the triadic relationships in which knowledge is imported into a discipline. Forced, because the relationships under investigation are often complex ones, much higher-adic-than-triadic. As the model developed for analysis here rests on Richmonds cyclic ideas, it may also be applicable to analyses involving a greater part of the immense complexity of interdisciplinary reasoning, but no such attempts will be made here.

A few comments on similarity

In many theories of knowledge production, the concept of likeness or resemblance plays a prominent role. This aspect of reasoning is particularly interesting when thinking about how knowledge from one domain may contribute to the production of knowledge in another. Likeness does play a role in most forms of interdisciplinarity, and I will give some comments on the subject in the following.

One way of thinking about likeness is analogy as a form of reasoning, as in Peirce: "Analogy is the inference that a not very large collection of objects which agree in various respects may very likely agree in another respect. For instance, the earth and Mars agree in so many respects that it seems not unlikely they may agree in being inhabited" [CP 1.69]. But also more contemporary writings have asserted that similarity plays some role in the selection of existing knowledge to be applied to new knowledge or observations. According to Lakoff & Johnson [1980] metaphor is at the core of all reasoning. Much of the body of metaphor theory puts similarity or likeness at the core of (metaphorical) reasoning. Way [1991] suggests that interpreting a metaphor involves an upward search through the hierarchy for a common ancestor of the two concepts involved. Generally, knowledge representation as Ontology often emphasises this kind of reasoning by generalisation. Views such as these may be (rightfully) opposed from many angles. For example, in prototype theory (following Rosch [1973]), concepts that do not share any features might still be in the same category. This presents obvious problems with establishing similarity by mere generalisation in type hierarchies.

A hybrid between these two views is Conceptual Blending Theory (see, for example Fauconnier & Turner [2002]). On another occasion [Hove & Schärfe 2005], we have argued that establishing similarity in the blending network might happen due to concepts presenting themselves as icons. An icon is "a sign which refers to the Object that it denotes merely by virtue of characters of its own, and which it possesses, just the same, whether any such Object actually exists or not. It is true that unless there really is such an Object, the Icon does not act as a sign" [CP 2.247]. So an icon stands for something else only because of its qualities. Qualities may be any feature of the sign. The icon then stands for its object because of a similarity in qualities (in at least some of Peirces writings). We then proceed to argue that this similarity may set off reasoning of any of the three forms: deduction, induction or abduction. Likewise, in the model to be presented here, the establishment of similarity may happen in both first-, second- and thirdness, thereby setting of any of the vectorial operations.

Although iconicity is often associated with firstness, the processes initiated by likeness need not be initiated in firstness. Iconicity does not necessarily play a role in the (vectorial) reasoning process, but in my model it seems to do so in establishing the likeness that serves to initiate the vectorial operations. I will not, however, go further into this question.

The discussion of similarity is interesting indeed, but it provides problems enough to be a masters thesis in its own right, and this is not the place to unfold it. Apart from that, it is my view that the model I will present is open to other notions of similarity, and that it would stand if other such theories were to be attached to it – for example the complete theory of Conceptual Blending, or a more complete version of the different signs in Peirce. In the analysis, I will adopt a simplified notion of similarity; I will identify features that are shared between the scientific disciplines in question, although these features, and thus the similarities, may be of a different nature.

Chapter 3 - Application of |>*

I am now ready to proceed to the actual application of the juxtaposition of theories outlined above. I will begin by commenting on the status of the analysis and transferability of the model. I will then proceed to introduce the scientific domain that will serve as my case in the analysis: Human-Centered Informatics. Then I will present the model and go into the actual analysis.

Status of the model and analysis

The model I am about to present is an abduction as to how ideas move into an inherently interdisciplinary science. The idea for the model has sprung from the earlier mentioned preliminary analysis of the interplay between Cognitive Science and Human-Centered Informatics, during the preparation of which I came to realise that interdisciplinary transfers between the disciplines happened in different realms, or at different levels. Out of this came the idea of a peircean semiotic analysis of Human-Centered Informatics as an interdisciplinary field. When reading through the literature of the field from a peircean point of view, it seemed to me that the way in which the idea or motivation for interdisciplinary transfer arises, might happen due to inadequacy, tension, likeness or some new idea in any of the three universes of science. In this respect and others, I found Richmonds vectorial analysis fascinating. When employing this theory, interdisciplinarity takes on a different form: Instead of viewing interdisciplinarity as "theories from two or more disciplines providing different views on the same object", or "theories and methods from one discipline employed to study problems of another", interdisciplinarity on the Richmondian view becomes a special process of reasoning, in which elements from foreign theories play a role in another. In other words, the idea for the model has come out of studying literature from the area of Human-Centered Informatics. The analysis is not intended to prove the model although it does serve to qualify it somewhat. What it is intended to do, is firstly to provide examples that these six modes of interdisciplinarity can in fact be identified within the setting of Human-Centered Informatics. Secondly, to assess how the model might provide clarity to the meaning of the concept interdisciplinarity in the context of this particular academic field – and how this clarification is useful. Thirdly, it provides some insights about the field of Human-Centered Informatics in itself. Because of the relative formality of the model of analysis, it can help clarify my original idea: that interdisciplinary import in Human-Centered Informatics happens in different realms. This should give some insight, although by no means exhaustive and not even sufficient, into how the field has evolved by drawing upon other fields. In this respect, it may be seen as an expansion of Øhrstrøms [2003] idea that Human-Centered Informatics is theories and methods from the humanities applied to problems from the natural sciences, and vice versa.

Transferability of the model

The study in which I will employ the model in the following is a local study, in the sense that it is occupied with the field of Human-Centered Informatics. The triad of science domain/analysis/theory> is not specific to Human-Centered Informatics, but covers science generally. In other words, it is open to attempts for application to other scientific fields. However, there is a number of reasons why Human-Centered Informatics seems well suited for analysis using this model. First of all, it is a relatively well-established field. Secondly, and more importantly, it is interdisciplinary in its very nature. As I have argued above, it originates out of a number of different traditions spanning facultary divisions. This is illustrated in the very different approach by the four fields of the science to the subject matter. The differences between the four areas with regards to theory, methodology and problems play a large part in driving the continued development of the field of Human-Centered Informatics. Thus, the model might be particularly well suited for this field. There are simply a lot of interdisciplinary import into it, and it is a scientific field with lots of intrinsic tension in all universes of the triad of science. As we have seen Øhrstrøm suggest, it contains theories and methodologies from different branches of the humanities as well as from the natural sciences, and these are applied to problems from both facultary areas. 25 years of research has caused for these theories, methods and problems to become intertwined, in a way that makes it difficult to distinguish between their origins. They have become theories, methods and problems of Human-Centered Informatics, and new elements from very different sciences keep making their way into the field. In this way, Human-Centered Informatics as a field lends itself particularly well to the model of analysis employed. As such, it may be an easy choice for qualifying the model. On the other hand, Human-Centered Informatics is a good example, since the model might be particularly fruitful for this scientific area. The main point here is, that while the model is sufficiently general that it **might** be applicable to other sciences, Human-Centered Informatics is probably a particularly good fit for it. In the conclusion following the analysis, I will reflect upon the applicability of the model to other areas of science.

Human-Centered Informatics – The domain for application

The viewpoint from which I will do my analysis is that of Human-Centered Informatics. This is chosen, in part because it is my field of study, but mostly because it is a good and typical example of a science that is interdisciplinary in the sense that it exists in interaction with a number of different 'traditional' sciences.

> "Human-Centered Informatics has a particular humanistic profile, placing human beings at the center of IT-processes. Science and technology are regarded as communicational, linguistic, ethical, and learning processes that create new knowledge, rather than as the simple invention of new technologies as such. The interdisciplinary research and educational programs cross the boundaries of the traditional academic disciplines and make these disciplines interact."
> [Clausen 2007]

Human-Centered Informatics may be seen as a typical and good example of something that is a scientific field in its own right, but one that lies (and prides itself in doing so) in between a number of different scientific fields. It is usually divided into four fields:

ICT, Learning and Cooperation

Design and Systems Development

Human-Computer Interaction

Knowledge and Formalisation

> “As a whole, Human-Centered Informatics comprise four distinct, yet related, branches of research(...)The four fields are independent in the sense that the researchers to some degree draw upon different theoretical positions and traditions, and participate in different national and international contexts of research. Such are the for research. On the other hand, it is quite obvious that the the fields can enrich each other; and when they are placed under the same heading – Human-Centered Informatics – it becomes almost self-evident that the four fields are closely related” [Schärfe 2003, my translation].

Even in terms of its own identity, the field is perceived to be interdisciplinary in nature. The division into four different fields means that Human-Centered Informatics has inherent internal tension. This tension, however, is also the driving force behind the science. Seen in a broader context, the interdisciplinary nature of Human-Centered Informatics becomes even more evident. As noted by Schärfe above, the four fields rest on somewhat different traditions. The research in the field ICT, Learning and Cooperation at Aalborg University is centred around the Centre for User Driven Innovation, Learning and Design, which, according to its website, “draws upon theory and methods from pedagogy, technology, organization, and design.”, theories traditionally belonging to very different traditions. They even cross facultary boundaries with pedagogy belonging to the humanities, technology belonging to the natural sciences, and organization belonging to the social sciences. In the light of facultary divisions, design is itself interdisciplinary, having one leg in humanities (aesthetics etc.) and the other in the natural sciences (such as architecture and industrial design). The same may be said about the other fields: Design and Systems Development drawing upon design as well as Computer Science and Organisation. Human-Computer Interaction which studies Computers from the viewpoint of for example psychology, and Knowledge and Formalisation, which is characterised by an interest in classical philosophy and Ontology. Within the field, many areas of interest may be identified, which cross the boundaries of the four areas, such as artificial intelligence, an example of a discipline that involves all the areas.

Choice of material for analysis

Since the focus in this thesis is on interdisciplinarity, the main principle in selecting objects of analysis is that they should be 'where interdisciplinarity is'.

An interesting feature of Human-Centered Informatics is its normative nature. In the above citing, it is explicitly stated as a goal for the discipline to 'make disciplines interact'. The fact that it is an actively interdisciplinary science in the sense that it seeks to cultivate interaction between fields. This, in turn, reflects on the literature that contributes to the formation of the field. By looking through the literature that is presented to the students of the field, I have made a selection that seems representative of one certain aspect of Human-Centered Informatics; the consciously interdisciplinary research. Although its scope and purpose may seem quite clear, Human-Centered Informatics is a diversified field, contributed to by many different scholars, and it would be impossible to incorporate enough literature in my analysis that it would be representative of the entire field. That being said, I have tried to pick examples that represent the breadth of the field, and the analysis covers all four areas of Human-Centered Informatics. Apart from a wish to capture the breadth of the fields, the main criterion for selection has been that the ideas chosen for analysis should be ones that have quite obviously been imported into Human-Centered Informatics from other traditional fields of science.

The model

In the preceding chapters, I have introduced:

- My conceptualisation of science into domain, analysis and theory (corresponding to firstness, secondness and thirdness)
- The idea of vectorial movements as presented by Richmond [2005, 2006]
- My conceptual modification of these movements into initiation, process and product
- The chiral cycle as presented by Richmond, while I tentatively suggest that cycles might be initiated in universes other than that of firstness.
- My adoption of a simple notion of similarity
- Human-Centered Informatics as a case

Having there prerequisites in place, I will present a model of interdisciplinarity. It stands as a postulate, but in actuality it is the product of the analyses to follow. Presenting it first is a matter of presentational tactics; I hope that knowing the part of the point in advance will make my line of argument easier to follow.

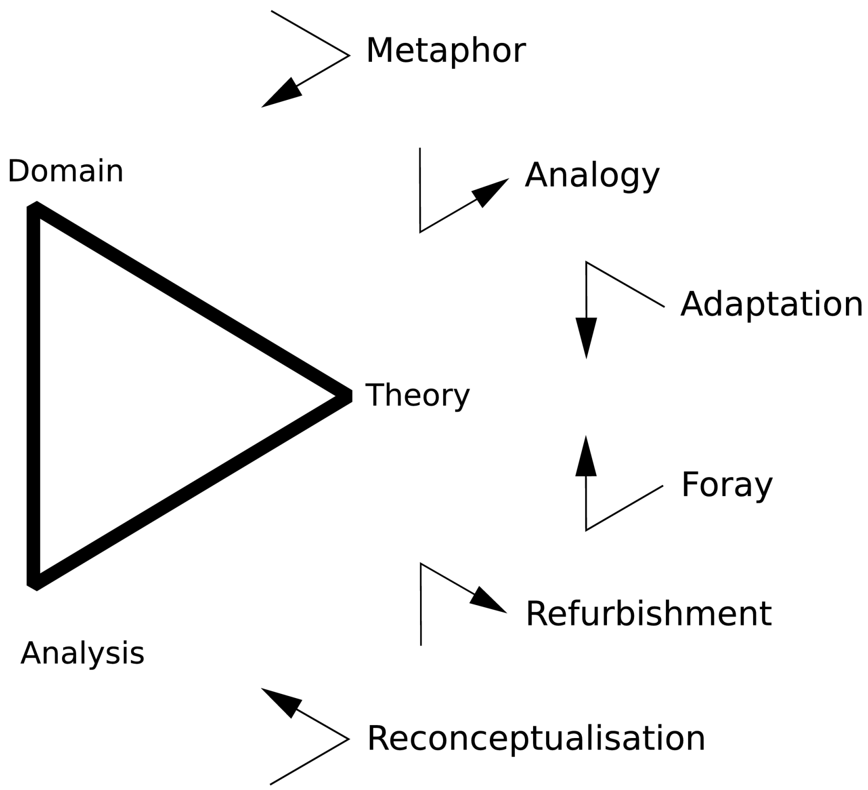
The chiral cycle covers 2 of the 6 different vectors. I suggest that the remaining 4 vectors may be split up into vectorial operations in the same way Richmond has done with the others.

The chiral cycle may be seen as idealised knowledge production rather than an exhaustive account of every kind of knowledge production. Apart from this, I have two further arguments for why the four remaining vectors could undergo the same treatment.

I suggest that there are six modes of interdisciplinarity. I arrive at this figure through a twofold line of reasoning:

1. A mathematical argument: accepting the notion of clockwise and counter-clockwise chiral cycles, and suggesting that such cycles may be initiated in either domain, analysis or theory, gives 6 possible combinations.
2. An empirical argument: Long before the conception of this somewhat formal model, I identified movements in Human-Centered Informatics literature that were initiated differently and followed different patterns and cycles of movement. The remainder of this report will be dedicated to making this part of the argument, as well as the argument that this line of thinking is useful in describing cross-disciplinary movements.

The six modes of interdisciplinarity are:



Examples of the different modes

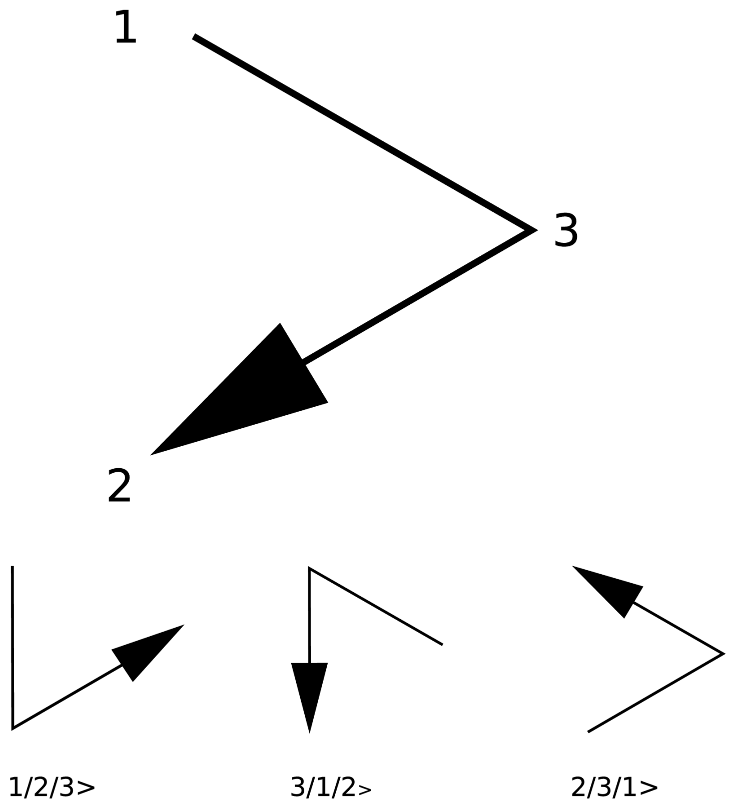
Reading Guide

As noted above, the vectorial analysis that will be applied to examples from Human-Centered Informatics in the following, may be seen on two levels. That of the overarching vector, and that of the vectors that make up the overarching vector, the sequential view. While these are two different representations of the same thing, they support two different ways of thinking about vectorial movements, both of which are fruitful:

When thinking of a cross-disciplinary process as **one vector**, such as metaphor 1/3/2>, the one-vector representation makes it easy to see how this is a movement from firstness (it is initiated, or driven, by the domain of by an idea), proceeding to reason in thirdness, the universe of theory, in order to produce re-analysis, secondness. The one-vector view is probably the more supportive in understanding the overall nature of a particular form of interdisciplinary import.

The sequential view is probably the more supportive in understanding how the actual import happens, as well as in reading the text of the analysis. In the illustration preceding every mode of interdisciplinary import, the one-vector view is broken down into a sequential one. In the case of metaphor, 1/3/2>, it breaks down as follows. The movement from 1 to 3 happens by 1/2/3>, that from 3 to 2 by 3/1/2>. Finally, the cycle returns to 1 by 2/3/1>, addressing the problem that set off the cycle. The sequential illustration also shows where the import happens: since it begins with 1/2/3>, the first process, and thus import, is in secondness, so import of practice or method. The following vector (3/1/2>) imports in firstness, and finally, 2/1/3> closes the cycle by importing in 1ns, so import has now happened in all universes. The sequence and initiation point of the vector is significant because the product of one vector becomes the initiation of the next; earlier vectors determine later ones. I will elaborate on this last comment after the analysis.

Software as Text – a case of metaphor



The overarching vector in metaphor is the vector of process. So metaphor is reasoning from likeness in the domain (1ns) through theory (3ns), to a new analysis (2ns).

Inger Lytjes Software as Text [2000] is a wide-ranging work, yet one with a clear focus: To reconceptualise systems development processes. [Hasle 2003]. Systems development is here to be taken in quite a broad sense, since the book is occupied with a relatively broad range of software-related issues: User interfaces, databases, knowledge bases and development of software. But, as noted above, the focus is clear throughout these diverse analyses, as is Lytjes way of making arguments: Different aspects of software is reconceptualised and analysed as text. The reconceptualisation (different way of seeing the domain) is the main point of the book, and is represented by the entire vectorial cycle seen above, illustrated by the fact that it begins as well as ends in the domain. The particular way of analysing software 'as if it was' a text makes this work a good and clear example of what I call metaphorical application.

1/2/3> Establishing similarity

The first cycle in metaphorical application is initiated in the domain. Part of the reasoning involved with metaphor as it is presented by e.g. Lakoff & Johnson [1980], is that one concept shares part of its characteristics with another. The same line of reasoning come into effect with domain in metaphorical application. The process is initiated by the fact that two domains share some characteristics – but like in conceptual metaphor theory, they do not have to share all characteristics in order to establish a metaphorical system of reasoning. This establishment of similarity is quite evident in Lytjes text:

> "I came to realise, that the processes beginning with a conception of software in use(...), to a surprisingly high degree bear similarities to the processes involved in the production and publication of books. I got the idea that the systems developer could be seen as a writer, and the user as a reader" [Lytje 2000, p. 5, my translation]

At first glance, this may seem like likeness in analysis rather than in domain. However, what Lytje emphasises is the conception of software, rather than the actual practice of developing or analysing software; An epistemological likeness between an author and a developer, a reader and a user, which are features in the domains rather than in the analysis of them. One example of such a feature pointed out by Lytje is that “software [kan] forstås som et dynamisk, semiotisk produkt, der er kodet i et bestemt tegnsystem” “software [may be] seen as a dynamic, semiotic product, which is coded in a certain system of signs” [p. 26 – my translation]

The similarity pointed out in the above citing leads to the inference that the analytical practices involved with text, that is both reading and writing, may be used for reasoning about software: She takes an interest in the concept of text, looks at what a text is, and what practices are involved with analysing texts. I take this to be an import of analysis, rather that one of theory. Because the domains, in spite of their claimed epistemological similarities, are quite different. The theories of semantic units, cognitive framing and metaphor are not immediately applicable to the domain of software, because these concepts and theories are tied to analysis of text. However, it is possible to look at the analytical practices in which the theories are employed, in order to set the background for analyses, to establish the metaphor. The metaphor, then, may be seen as a preliminary theory (3ns).

3/1/2> Expanding the metaphor

In the previous movement, the practice of making sense of the domain (text) was imported as a preliminary theory into the domain of software. One feature of a metaphor is that it lends itself to what has often been called 'expansion'. The idea is that since two domains are similar in one respect, they are likely to also agree in others (or to be reasoned about in the same manner). Once a metaphor have been established, it may be expanded by means of entailments. Lakoff & Johnson [1980] gives Love IS A collaborative work of art as one example – from knowledge of what it means for something to be a collaborative work of art, entailments arise, such as 'love requires compromise'.

In the same way, taking software to be texts has its entailments which expand the metaphor.

The vectorial movement begins with the 'preliminary theory' that is the metaphor (3ns). This movement then expands this metaphor by means of entailments. Entailments are actually import of problems from one domain into another – here, problems or epistemological characteristics (1ns) of text can be taken as characteristic of software, since it is done on the basis of the metaphor. Lytje chooses four different directions in textual analysis to be applied to software: cognitive linguistics, post-structuralism, pragmatics and generative linguistics. I will look at one way in which the perception of the domain of text (1ns) is imported as a way of perceiving the domain of software. As mentioned, one of the fields of textual analysis from which problems are imported, is cognitive linguistics.

In cognitive linguistics, at least according to Lytje and the sources she cites (e.g. Langacker) one basic premise of cognitive linguistics is that texts do not have their meaning objectively, rather, meaning is constructed during the process of understanding. So the meaning of a text cannot be derived from the text alone. One is able to read text written by others if, through grammatical analysis, one is able to construct a conceptual model representative of the text. [Lytje 2000, p. 30]. This basic epistemological entailment of cognitive linguistics expands the metaphor of software as text, in a way such that the meaning of software, in the same manner as that of text, is constructed during the process of understanding. And that it is understood through the construction of a conceptual model, without looking at which its meaning cannot be derived. This is also a hint at why this vectorial operation ends in 2ns; The problem imported from cognitive linguistics (the fact that meaning is constructed) calls for a new practice of analysis (one in which conceptual models play a role)

2/3/1> Applying theory as metaphor

The expansion of the metaphor outlined above imports new problems into the domain of software. This calls for a different practice in analysing software. In order to put this practice (2ns) to work on software (the domain, 1ns), an application of theories from cognitive linguistics (3ns) takes place – to the domain of software. Fortunately, the metaphor established in the first vector and expanded in the second, allows this. Theories from textual analysis, in this example cognitive linguistics, is applied to software as if it was a text. In this final phase, the actual import of theory happens (the process happens in thirdness). As suggested in the previous vector, the theory imported must address the issue that software must now be understood by looking at conceptual models. One of several theories that Lytje imports in order to address this issue, is the theory of cognitive 'frames'. Here she draws upon Fillmore [1982] and Talmy [1996].

One case in which Lytje applies the theory on frames is in an analysis of the graphical user interface of the e-mail client *Netscape Messenger.* Here, she makes the observation that the interface expresses a metaphor[3] by analysing it in terms of a cognitive frame:

> "Kildedomæne er post i traditionel forstand omfattende postkasse, postbud, brev etc. (...) Breve bliver lagt på et bestemt sted, nemlig i en postkasse, og derfra hentes de for at blive transporteret til en anden postkasse, hvorfra det igen bliver hentet etc. for til sidst at blive bragt til modtageren (...) Dette billede af post er i høj grad reproduceret i

3 There is imminent danger of confusion of concepts here. It is important to note, that in this passage, Lytje is applying metaphor as part of what I call the metaphorical application. The two levels are not to be confused.

> messenger" "Source-domain is mail in the traditional sense, including mailbox, mail man, letter etc. (...) Letters are put in a designated place, a mailbox, and from there they are collected for transportation to another mailbox, from which it is then collected etc., and in the end they are finally delivered to the recipient (...) This image of mail is reproduced to a high degree in Messenger" [p. 97.]

She reaches this conclusion by analysing Messenger as if it was a text requiring a conceptual model on the part of the recipient in order to be understood. Like texts, from the point of view of cognitive linguistics, Messenger uses metaphor and frames recognisable by the user in order to make sense. It is interesting to note how this analysis leads Lytje to point out that the interface allows these frames and categories to drift. In the domain of physical mail, when a mailbox has been emptied, it is in fact empty. This is not the case with the inbox in an e-mail client. This is a very good example of how the metaphorical application of textual theories to software changes the perception of the domain (1ns, which is where the vectorial cycle of metaphorical application ultimately ends). Software not only draws upon conceptual models. Like a text, it also changes them.

The cognitive linguistics-example that i have tracked through the cycle above is one of the most trivial ones i could have chosen. I did so to be able to make the analysis as clear as possible. However, the application of Pragmatics and post-structuralism to software has deep implications on how software may be perceived. The following is a citing in which it is particularly clear that import of a theory on text (3ns) gives new insights into the domain of software:

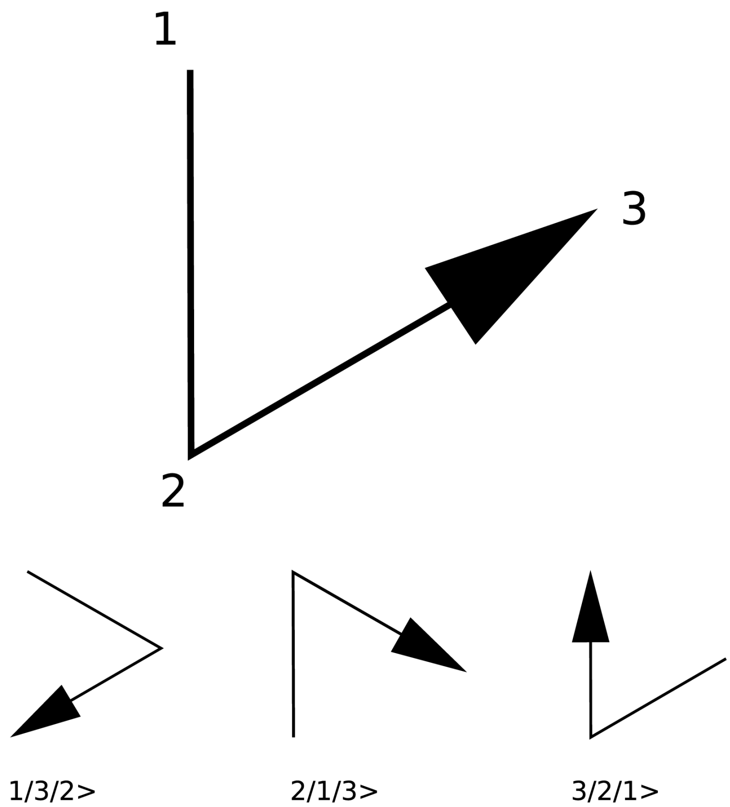
> "The enquiry gave gave rise to reflection pertaining to interaction, which, on my view, leads to the understanding that influence is two-way. The system influences the mental state of the user and motivates cognitive occurrences in the user, while the user influences the

> information available for one to come into contact with (...) through the mechanisms represented by the computer based sign. This definition is actually in concordance with Levinsons definition of interaction" [Lytje 2000, p.103]

The above constitutes the conclusion of an analysis where Lytje has examined a software interface from textual methods of analysis. This perception of humans and software as something that exists in mutual interplay is one of the core points of *Software as text.* It is possible to make the above argument, because the metaphor-expansion has allowed her to look at software from the point of view of the theories of Levinson, and through that to see problems in the relation between the user and software in a new way. Levinson states that interaction is on-going production of chains of mutually dependent actions, and by applying this theory to the metaphorical view of software development, this too becomes on-going production of chains of mutually dependent actions.

Birth of the Scandinavian tradition in systems development (object orientation) - a case of analogy

Analogy is a name for the clockwise vectorial cycle initiated in the domain:



The overarching vector in analogy is that of order. So it is reasoning from likeness in the domain (1ns), through analysis (2ns), in order to develop theory (3ns).

The term analogy has to do with the domains. It is a special kind of interdisciplinarity that happens when domains are analogous, i.e. they display similar problems. A cycle of analogy may be set off by some variation in the domain, as is the case with what has often been labelled 'The Scandinavian Tradition' in systems development. Through the seventies and eighties it became apparent, in Scandinavia in particular, that the introduction and development of information technology in the workplace often shows unfortunate implications for the people that use them. They become alienised with regards to their own jobs because the systems fail to relate to the practice in which they are used. One of the main pioneers behind the Scandinavian tradition and Object Orientation, Kristen Nygaard, mentions Operations Research (OR) as a major inspiration in the development of Object Orientation. OR as a science dates back to World War II, and was developed for military purposes, but after the war it made its way into business, industry and government [Churchman et al. 1957, p. 3]. As a discipline, OR is occupied with how resources may be utilized optimally for a given task. It is highly mathematical, but in this analysis, I will not go into these parts of the theory. Instead, I will look at the very interesting line of thinking that OR represents.

1/3/2> A change in perception of Information Technology

The likeness between OR and Systems Development emerged as part of a general tendency in society through the seventies and eighties, where “a growing awareness of the implications of computerization of work processes for individuals, organizations and society as a whole” became apparent [Lytje

1989]. This tendency was especially prevalent in Scandinavia, where strong trade unions are advocating the rights of the workers in the implementation of computer systems. We see here at least two areas of tension; there is tension between the analytical realm and the theoretical – the theory gives a conceptualisation of the domain that fails to function when put into practice, or at least, the systems developed by use of existing theories (the Peircean notion of 'habit' may actually be more accurate here, as early systems development was largely atheoretical) give unsatisfactory results when implemented. In pushing developers toward taking into account the welfare of the workers, and the need to avoid their alienation towards their jobs, they point out a tension between their theory (critical work research) and the domain of computing. This gives rise to the first vector in the cycle of analysis, 1/3/2>: The introduction of the computer into the workplace has caused problems to arise that cannot be analysed by means of theories from Computer Science nor theories pertaining to work. This variation in the domain of systems development calls for revision of analyses according to the new theory. This sets off a vectorial cycle:

The Simula languages emerge among the earliest examples of object oriented programming. Developed in Norway by Ole-Johan Dahl and Kristen Nygaard beginning from the sixties, it sprang out of Norwegian Operational Research [Nygaard 1992], and was intended to meet the "need for precise tools for the description and simulation of complex man-machine systems." . In his 1992 paper *How many basic choices do we really make – How many are difficult*, Nygaard sheds light over some of the motivations lying behind the development of the Scandinavian tradition in systems development. About his own perspective on systems development, he states:

> "For me informatics and OR have always been closely related, and I tend to see many tasks in informatics from the perspective of OR (...) A main and, at the time, largely undebated assumption in the development of the post-war culture was that "technological progress

> happens, it is politically neutral - and good!". (The concern about atomic weapons was one of the exceptions.) In Operational Research, however, the situation was somewhat different: The task was to find the best use of men and equipment, dependent upon a stated set of objectives. If the objectives were modified, the "best use" changed. Also, the development of new equipment had to be fine tuned to a proper understanding of the objectives of the decision-makers. And those objectives could be highly political, particularly in the military field. The application of OR techniques to conflicts between interest groups within organizations was an idea dear to an OR researcher.”
>
> [Nygaard 1992]

The paper gives a rare opportunity for a view into the choices of a scientist, where he explicates, not only his scientific commitments but also social and ethical ones. The last two usually require some interpretation to extract from scientific publications. For this reason and others, this paper is an interesting case for vectorial analysis. The above citing contains a number of vectorial movements

”technological progress happens, it is politically neutral – and good!” causes a tension with the fact pointed out by unions that technology is highly political.

In Operational Research, however, the situation was somewhat different: The task was to find the best use of men and equipment, dependent upon a stated set of objectives. If the objectives were modified, the "best use" changed.

Nygaard observes this through OR research, where the optimum use of man and machinery varies with the objectives of the optimisation. It should be noted here that Nygaard for personal ethical reasons chose to work closely with the trade unions, and had a bias towards work research. This and theory from OR allowed him to see a solution to the fact that the domain (1) has changed from 'machines as systems' to 'humans and machines as systems'. The solution being to look at the domain from the perspective of OR (3ns), since the development in the domain of systems development had caused it to share features with the domain of OR – man and machine is seen as one entity rather than two separate ones. From these theories a practice (2ns) of Systems development may be derived by importing theoretical elements from OR.

2/1/3> Developing a theory

This phase described above merely constitutes the inspiration for the application of new theory. Practices from OR have yet to be used against the domain of systems development. This happens through determination (2/1/3>), through which a theory of Systems Development is formed. The process of this phase happens in firstness, so it is import of epistemological elements from the domain of OR to that of systems development. In the following, I will give some examples of epistemological characteristics from OR that are assumed to be epistemological characteristics of Systems Development.

One of the problems that Nygaard identified as an inspiration for developing Object Orientation is that the employees are not taken into account in theories on Systems Development. One epistemological characteristic in OR is the wide concept of a 'system'. In an industrial context, such as the one OR is written into in Churchman et al. [1957], the system is comprised of components such as: "controllers, agents who carry out policies, instruments and materials used in so doing, outsiders who are affected by the organization's activity, and the social environment in which these components operate" [Churchman 1957, p. 109]. He proceeds to name examples such as management, men, machines, materials, consumers, competitors and the government. It is noted that these entities make up a system by virtue of organisation. This clearly illustrates how the organisation, the machines, the employees and a number of other phenomena is seen as one system. The domain of Systems Development then, might be assumed to also share these characteristics.

Another problem pointed out in the preceding vector is that in Systems Development, technology was largely seen as neutral. However, the effect on the employees told a different story. OR sees any analysis or model of something as depending on objectives. One piece of early OR literature, Churchman et al. [1957], refers to a phenomenon they call 'weighting objectives', in determining the weights given in the mathematical models:

> “The necessary relative weights might be assigned in terms of dollar amounts by merely putting a certain dollar sign on every objective. This has the apparent advantages of a measure that is readily understandable, “objective”, and universally used. The difficulties in the use of monetary scales are also apparent. Many objectives cannot be measured in terms of dollars. In many cases we value differently two things which can be obtained at the same cost. In other cases costs are very difficult to assign. For example, what is the true cost for an injury, a life, a failure to supply an item, a loss of “good will”? (...) while it is true that these objectives cannot be “touched” by a dollar method of measurement, they can nevertheless be measured by other methods” [Churchman 1957]

In part due to its mathematical nature, 'objective' means something slightly different here. It remains, however, that the outcome of any analysis is determined by the objectives. What is perceived to be part of the system is determined by objectives. If this is assumed to be the case in Systems Development as well, it becomes possible to take into account the 'objectives' of workers when analysing some problem for the design of a computer system.

When these problems from OR (1) are taken to be problems of Systems Development, this forms a new theoretical basis (3ns)

3/2/1> Re-interpreting the domain

The application and implementation of this modified theory of systems development brings about new possibilities in the domain, as 3/2/1> analysis. The possibilities themselves are difficult to observe (they are 'airy nothingness'), but Nygaard notes the consequences of them:

> "Simula I, was made available in the spring of 1965, it was immediately used in a series of jobs in Norway and, even more, in Sweden. It was of course fascinating to see the tool we had developed being put to practical use and influencing the design of organizations and production facilities." [Nygaard 1992]

This citing contains a number of vector operations in itself, but it shows that Simula I certainly had impact on the domain.

In order to show how the ideas tracked in the preceding vector have made their way into (relatively) modern day Object Orientation, I will turn to Mathiasen et al.'s *Objekt Orienteret Analyse & Design* (Object Oriented Analysis & Design (OOA&D)), to give a few examples of how practices (process in 2ns) from OR give rise to new possibilities in the domain of Software development.

That 'systems' are perceived as including, not only machines, documents, etc., but also for example workers, management and surroundings, has the implication that Systems Developers can no longer look at just the computer; The computer is only part of the system doing the work. The phenomenon which is called 'the system' in OR is called the 'problem domain' in OOA&D. The idea behind both is that any system/problem domain may be modelled, and that this model can tell something about what is modelled. The 'model', in the words of OR, is a "representation which lends itself to use in predicting the effect on the system's effectiveness of possible changes in the system". Similarly, in OR the model is designed from analyses of the problem domain [Mathiasen et al 2001., p. 230]

One of the key ideas of OR is that a system, such as an organisation, seen as interaction of functional units, may be modelled using various techniques. [Churchman et al., p. 155]. This idea also forms the core of OOA&D. The problem-domain, in the terminology of OO is: "Den del af omgivelserne, der administreres, overvåges eller styres ved hjælp af et system", "That part of the surroundings that are administered, monitored or controlled using a system" [Mathiasen et al. 2001, p. 6]. So the problem domain corresponds nicely to the notion of the 'system' in OR (note that Mathiasen et al. use the word 'system' to denote computer system, so differently from OR). Like in OR, from analysis of the model domain is derived a 'model component'. This model is of a different nature than the mathematical models developed in OR, but the techniques for developing the model in OR and OOA&D (the practices, 2ns) are quite similar. For example, the system/problem domain is analysed as components. In OOA&D, these components are called objects:

"We can begin to construct a symbolic model of the system by itemizing all the components of the system which contribute to the effectiveness or ineffectiveness of the operation of the system" [p. 163]. In OOA&D an 'object' is described as an abstraction over a phenomenon in the problem domain. Like in OR, the components (the objects in OOA&D) of the system (the problem domain in OOA&D) are itemized as functional units.

Common to OR is also the idea that phenomena in the problem domain should be accepted or rejected as objects depending on the actions they are involved in:

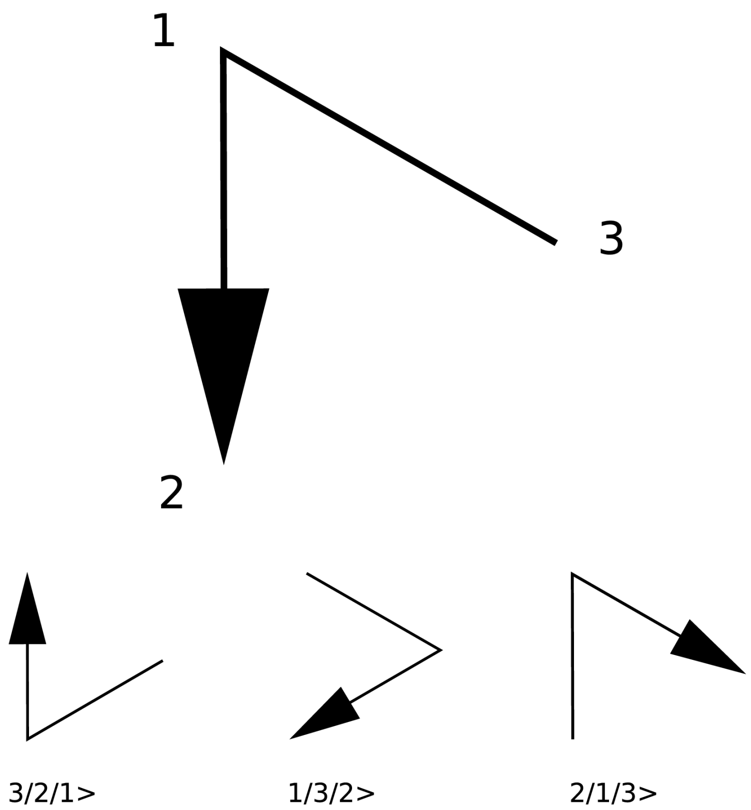
> "Once a complete list of the components of the system is compiled, the next step is to determine whether or not each of these components should be taken into account. This is done for each component listed by determining whether or not this component is affected by the choice of a course of action from among alternatives" [Churchman 1957]

The same principle is applied in OO - if a phenomenon is not involved in an action that somehow changes something in the problem domain, they should not become objects.

This final vector provides systems development with a new view of its domain, addressing the problem that set the cycle of analogy off to begin with: That technology can have unfortunate implications for the workers using them. In the first vector 1/3/2> OR is identified as a domain with problems analogous to systems development. In the second vector 2/1/3> epistemological characteristics from the domain of OR (1) are taken to be characteristics of Systems Development, enabling the development of a theory (3) that take these problems into account. In the third vector, by employing the practices (2ns) inspired by the theories from OR (initiated in 3ns), changes happen in the domain of systems development – through reinterpretation by 3/2/1>, a new view of the domain is developed: One in which systems are perceived as interconnected objects and where optimal solutions are dependent on the objectives of the system.

Acting with technology – a case of adaptation

Adaptation is the clockwise vectorial operation initiated by theory:



The overarching vector in adaptation is that of representation. It is reasoning from some theoretical likeness (3ns). Foreign theories are then applied to problems in the domain (1ns), giving new analyses (2ns) that show the consequences of these theories.

The volume *Acting with technology: Activity theory and Interaction Design*, [Kaptelinin & Nardi 2006] explicates as its three foci as: (1) What impact has activity theory had on interaction design?, (2) How does activity theory relate to other theoretical approaches in the field? and (3) What does "activity theory" really mean? As such, the book is not as much theory development as it is an analytical one. This choice on my part is deliberate: like in evolutionary biology, adaptation in the context of science is a process that evolves over periods of time. It is the process by which theories change due to broader academic or even societal movements (I'm avoiding the word paradigm here). Adaptation is thus a response to changes in the theoretical framework that sciences draw upon. Tendencies toward doing this is particularly evident in interdisciplinary sciences; theoretical tendencies, movements or revolutions in the theoretical basis that the interdisciplinary science rests upon somehow require a response, or at least that the science somehow relate to them. I will take an interest mostly in the first focus of *Acting with Technology*, i.e. the retrospective inquiry into how Activity Theory (a broad movement stemming from psychology) has influenced Interaction Design. For the purpose of their analysis, Kaptelinin & Nardi define Interaction Design as "all efforts to understand human engagement with digital technology and all efforts to use that knowledge to design more useful and pleasing artefacts.". From this definition, it is clear that Interaction Design as a science falls within the scope of HCI. I will identify the vectors by which the relevance of a theoretical movement in some field is recognised as relevant to Interaction Design, imported and made into HCIs own.

3/2/1> Responding to theoretical movements

This initiating vector, 3/2/1> is very evident in the following citing:

> "Activity theory [3ns] opens up avenues of discussion [ending in 1ns] concerning human interaction with technology and potentially can be fruitful in encouraging participation in conversations about the larger global concerns that the deployment of our technologies unquestionably affects. If we are acting with technology [i.e. employing the practice of Activity Theory, import in 2ns], both possibilities and responsibilities expand." [p. 26, my comments in brackets]

What the citing does not show, however, is the initiation of the vector in thirdness. It shows that it is Activity Theory that opens the discussion, but not why this particular line of theory is chosen. One explanation might be found in the interdisciplinary nature of HCI, including Interaction Design. HCI as a discipline arose in the early 1980s from a number of different disciplines, among these cognitive psychology and computer science. [Carroll 2003]. HCI adopted the information-processing paradigm of computer science as the model for human cognition, and this 'mind-as-machine' perception developed in HCI has influence back on to the field of cognitive science. Such cognitivist approaches, however, has been questioned by a number of different developments within psychology – one of these being Activity Theory. The main point here is that the sciences of HCI and (cognitive) psychology share considerable part of their theoretical basis; the sciences may be seen as somehow existing in parallel, or in mutual interplay. So, when there is stir in the theoretical basis of cognitive psychology it is natural that researchers within HCI feel that they are required to respond or relate to these developments. So, Kaptelinin & Nardi make the observation that 'something is happening' in the theories of psychology:

> “Today activity theory is an approach that has transcended both international and disciplinary borders. It is used not only in Russia, where it originated, but also in Australia, Belgium, (...) and other countries.” and they proceed into the next vector: “It is applied in psychology, education, work research, and other fields.” [p. 6]

'Adaptation' is one way of making such a response. In vectorial terms at least, this way of relating to theoretical developments is fairly straightforward: The new theory from psychology, Activity Theory, is applied to the domain of Interaction Design, by analysing this domain using the analytical practices (2ns) of Activity theory. In this first vector of the cycle, the process (and the import) happens in 2ns, the analytical practices. On the most general level, the analytical methodology of Activity Theory is characterised by the fact that is “aims to understand individual human beings, as well as the social entities they compose, in their natural everyday life circumstances, through an analysis of the genesis, structure, and processes of their activities.” [p. 31]. This way of analysing ones domain is special in several ways, and as such has several implications for the problems (1ns) observed in ones domain: It does not only analyse individuals, but also “the social entities they compose”. This is a considerable expansion of context compared to cognitivist approaches, since it is no longer possible to conduct analyses by looking at the individual; One must also consider the social context that the individual is part of. This way of analysing has a number of methodological implications: Analysis is occupied with 'natural everyday life circumstances', as opposed to strictly theoretical investigation, and even laboratory experiments. The key focus when looking at ones domain is “activities”, meaning not only human activity but “activity of any subject, [which] is understood as a purposeful interaction of the subject with the world, a process in which mutual transformations between the poles of “subject - object” are accomplished [Kaptelinin & Nardi 2006 citing Leontiev 1978]. So when analysing the domain, not only humans can act, any subject, including artefacts, are seen as actors engaged in activity. An activity that is interaction with the world, serving some purpose, introducing purpose as something to look for when analysing. Finally, interaction results in 'mutual transformations'. It is obvious that this particular method of analysis 'opens avenues of discussion'. This vector has its product in firstness, a new epistemology in which existing problems in the domain of Interaction Design is seen in new way, and in which new problems may be encountered. Kaptelinin & Nardi [2006] give a number of examples of such new ways of seeing the domain, one example:

“according to activity theory, “user - system” interaction is too narrow a phenomenon to count as a genuine activity. Making a meaningful activity the unit of analysis means that not only an interaction between people and technology is considered, but also the objects in the world with which subjects are interacting via technology.” [p. 34]

1/3/2> Applying the theory to HCI

When Interaction Design, as a domain (1ns) presents itself in a different way (this vector is initiated in firstness), this calls for a revision of the analytical practices of the science (2ns, the product of this vector). This is where the actual theoretical application takes place (process in 3ns). While the initiating vector 3/2/1> identified new and different problems in the domain, this vector puts the theories from Activity Theory into a practice applicable to HCI. The inference by which this is done is 1/3/2>: How may the new problems encountered in Interaction Design (1ns), be addressed by the theories of Activity theory (3ns), so that a new way of analysing (2ns) Interaction Design may be developed?

Beginning in 1ns, this vector takes the viewpoint of the domain in question, Interaction Design, in order to find theoretical elements in Activity Theory that might lend themselves to application to Interaction Design. 'Mediation through artefacts' is a theoretical element of Activity Theory that springs from the fact that not only humans but also artefacts qualify as subjects involved in some activity. Kaptelinin & Nardi give the following interpretation (in their condensed overview of Vygotskyan Activity Theory, i.e. not in the context of Interaction Design):

> “Human beings seldom interact with the world directly. An enormous number of artefacts has been developed by humankind to mediate our relationship with the world. Using these artefacts is the hallmark of living the life of a human being. Tools or instruments - physical artefacts mediating external activities - are easy to recognize, and their impact on the everyday life of every individual is obvious.” [p. 42]

When reading the above from the viewpoint of Interaction Design (or, for that matter, HCI more broadly), the applicability to the domain is quite clear; Computers are such artefacts. So at least two theoretical notions can be imported in the design of an activity theoretical methodology for the analysis of Human-Computer Interaction: (1) Computers are physical artefacts mediating external activities and (2) Computers are objects which have 'purposeful interaction with the world'. Also, keep in mind the import from the analytical methodology of Activity Theory that the unit of analysis is not the objects (e.g. the humans, or the computers), but rather 'meaningful activity', i.e. the purpose of the interaction.

As such, a framework for analysis is developed, which is ready for application to the domain of Interaction Design.

However, having an analytical framework in place does not constitute a useful theory of interaction design. Kaptelinen & Nardi define as the role of Activity Theory in interaction design to “reframe key concepts” [p. 73]. This is done by re-analysing the 'key concepts' of interaction design (process happening in 1ns), using the analytical framework (beginning in 2ns) imported from Activity Theory. Through the use of the method of analysis developed in the preceding vector, epistemological features of Interaction Design changes through re-analysis, i.e. certain problems (1ns) from Activity Theory are imported to become problems of Interaction Design. Bødker [1989], a pioneer in the application of Activity Theory to Interaction Design, writes the epistemology of Activity Theory into HCI as follows:

> “Computer-based artefacts often allow no direct access to the subject or object of the actions conducted through the artefact (e.g., a word processing program). Sometimes the object does not even exist as something separate from the artefact (e.g., a spreadsheet). It is part of human capabilities that make us able to project our experiences with one object onto another object (e.g., a blind person who is using a stick to "seen the surroundings).(...) The user interface forms the artefact-bound conditions for how actions can be done. These can be conditions for operations directed toward the artefact and for operations directed toward the real objects or subjects at different levels.” [Bødker 1989 pp. 178, 180]

This revision of HCI problems is the first step in the constitution of a Activity Theoretical theory (3ns) of HCI. But in order to fully flesh out the theory as one of HCI, re-analysis of the domain is called for. In order for something to be a theory of Interaction Design, it is obvious that it has to be concerned with the latter – design. Activity Theory in psychology is not a constructive theory as such, but its analytical framework may be applied to the problems of design, in order to move towards a theory of design than honours the epistemology and analytical practices of Activity Theory.

> “Considering information technology a mediating artefact, rather than merely a pole of human- computer interaction, has a straightforward implication for design. If taken seriously, this notion requires that the most important design objective should be to help people attain their meaningful goals. However obvious, this objective is seldom an explicit concern for designers of digital work environments.” [Kaptelinen & Nardi 2006, p. 117]

The above illustrates this final vector quite clearly. The analytical practice of identifying mediating artefacts is applied to HCI so that information technology may be considered such a mediating artefact. HCI, then, inherits some problems from Activity Theory (import in 1ns): Artefacts should 'help people attain their meaningful goals'. Since computers and user interfaces are such artefacts, this purpose holds for those as well. So the application of analytical methods from Activity Theory has *implications* for design. The methods (2ns) are used against the domain of Interaction Design (1ns), and through this, a theory begins to develop (3ns). So, 2/1/3>.

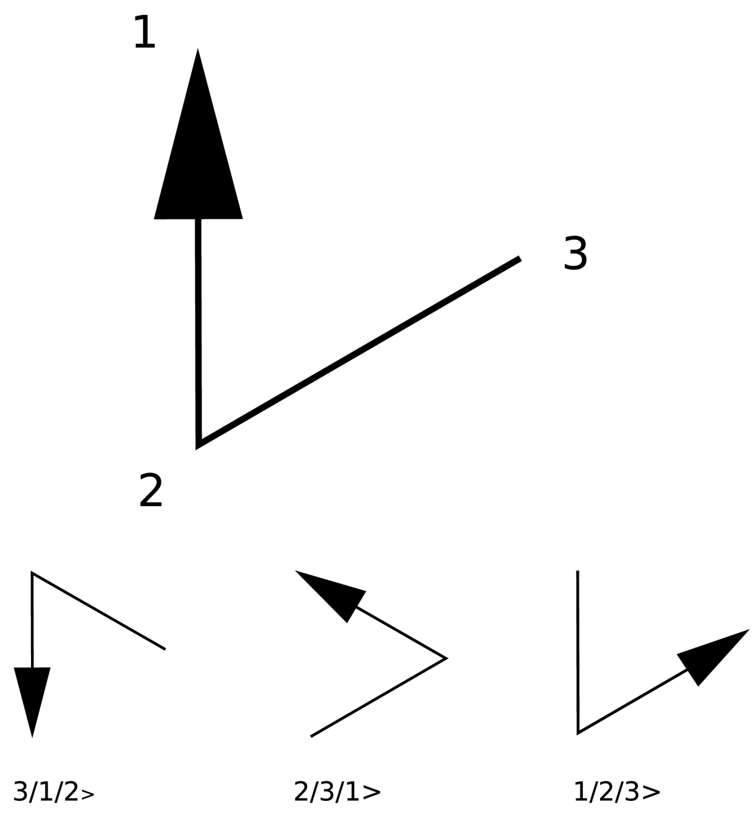
An example of one such theory is Bødker [1989], a condensed version of her doctoral thesis *Through the Interface – A Human Activity Approach to User Interface Design.* In her thesis, she develops a framework for the analysis and design of user interfaces, with Activity Theory as the main (though not the sole) theoretical basis. One of the main conclusions of this work is:

> “To design an artefact means not only to design the artefacts for a specific kind of activity. Because the use of artefacts is part of social activity, we design new conditions for collective activity (e.g., new

> division of labour and other ways of coordination, control, and communication)." [Bødker 1989, p. 193]

This conclusion clearly shows how the new role of artefacts (as part of activity), has deep implications for design. Artefacts cannot be understood detached from activity, and they are not designed 'for a specific kind of activity'. Perhaps the most important contribution of this 'adaptation' of Activity Theory to HCI is this. The objects of analysis changes from "user – system" to meaningful activity mediated by artefacts (i.e. computers). Design is not determined by an activity, rather, the design of artefacts is also the design of practice, and design and practice exist in mutual interplay.

Parallel Distributed Processing – a case of foray



The overarching vector in foray is that of analysis. This line of reasoning begins from a theoretical likeness (3ns), and the proceeds directly into analysis (2ns), in order to derive another conceptualisation of the domain (1ns).

The word foray is meant to picture the theories of one science venturing into a domain other than the one for and through which they were developed. A foray is initiated when it seems possible to describe two domains using the same or similar theories. The hypothesis is, then that the domains themselves also share some features. One example of foray is Neural Networks. PDP belongs to a special, somewhat separate, branch of Cognitive Science as well as Artificial Intelligence, often referred to as connectionism.

Connectionism is particularly interesting, because it is a line of theory that is developed, not through the study of mind nor through the study of Artificial Intelligence, i.e. computers, but in an interplay between the two sciences. (The relationship between Cognitive Science and AI is highly complex, and I am taking the liberty of merely postulating it here. One attempt at characterising this relationship is Bechtel & Graham [1998], a work much too comprehensive to even outline here.). Roughly described, this means that insights gained from studying mind may be applied to AI, and vice versa, and that the theories applied may be applied to both domains.

As an example I will use one particular branch within Neural Networks, called Parallel Distributed Processing (PDP). The theories on PDP are complex and highly technical, so in this analysis I will intentionally subject them to gross oversimplification, and keep with the general idea of the theories. For this reason, I have chosen a simplified account of PDP as the object of analysis: That from Copelands 1993 *Artificial Intelligence.*

3/1/2> - A unity of theory

The cycle of foray is initiated by a theoretical likeness. However, differently from the cycle of adaptation, in foray, the theoretical likeness is not an established one. Instead, it is merely postulated, hypothesised or guessed. In adaptation, the process happens in secondness, and is one of analytical practices. As such, the initiating theory determines the perspective of analysis, because its methods are applied to another domain. Not so in foray, since the process happens in firstness: Due to their theoretical likeness, it is assumed that the domains themselves share characteristics. One such hypothesis of likeness is the Symbol Systems Hypothesis, a term coined by Newell & Simon [1976].

> “The symbol system hypothesis implies that the symbolic behaviour of man arises because he has the characteristics of a physical symbol system. Hence, the results of efforts to model human behaviour with symbol systems become an important part of the evidence for the hypothesis, and research in artificial intelligence goes on in close collaboration with research in information processing psychology, as it is usually called.” [Newell & Simon 1976]

This is an example of the view in AI that the mind may be modelled as a computer, and vice versa. Although it is not directly connected with connectionist theories such as PDP, it serves to illustrate the line of thinking in the AI-traditions which assume the domains of mind and machine to be essentially the same, in the sense that they are capable of thinking due to their being a symbol system. Hypothesis such as this constitute the theoretical likeness, together with the fact that many scientists within the AI community want minds and computers to do essentially the same thing. From this hypothesis (hypotheses may be perceived as theories at an early stage, so 3ns), it may be inferred that there might also be a likeness in domains (1ns) This inference serves as justification for the hypothesis that the domains intelligence in minds and intelligence in computers share similarities. The above passage contains a comparably large numbers of may, mights and hypotheses. This is no coincidence, since 3/1/2> corresponds to the abductive form of reasoning, also illustrated by the fact that this vector ends in 2ns, brute force, i.e. 'trying something in practice'.

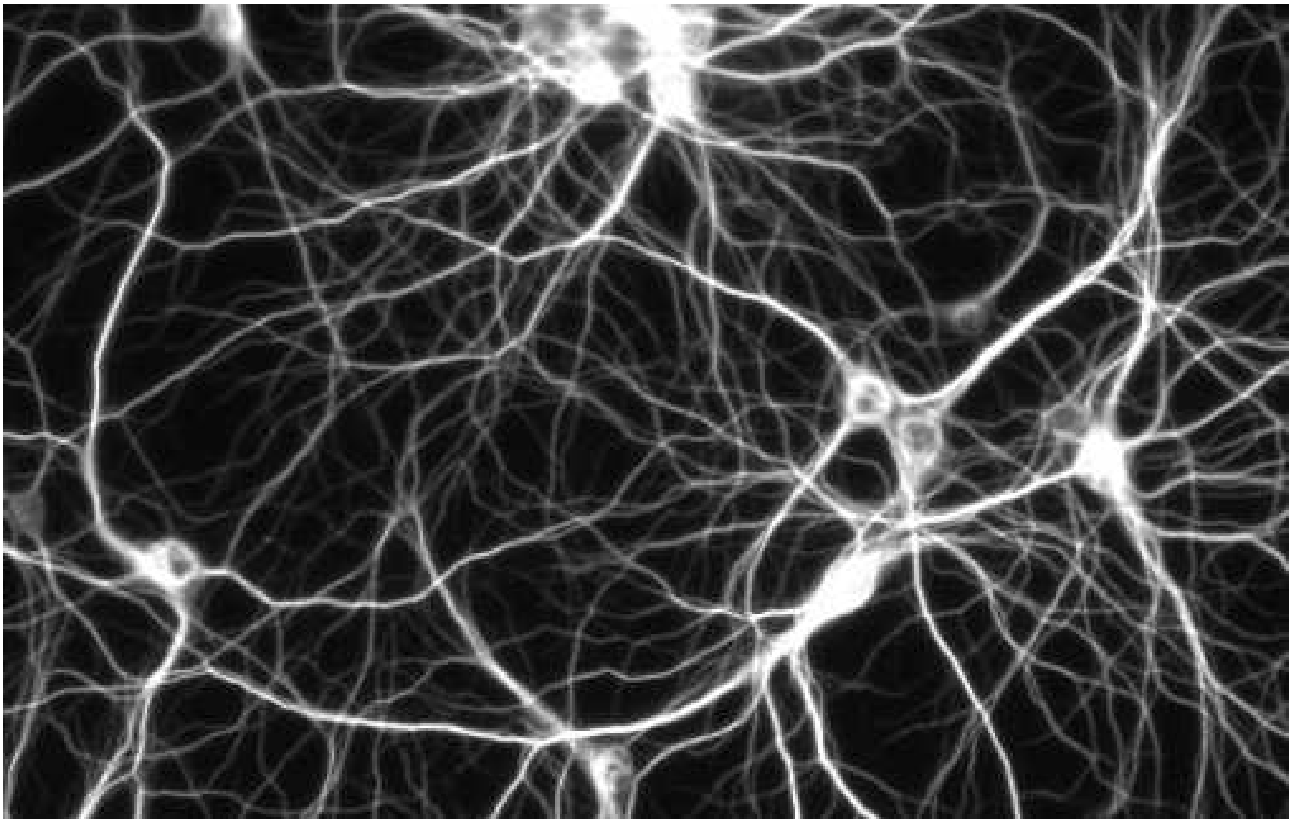
The process bears some similarity to metaphorical application; it consists of the same vectors. The only difference is in the sequence. However, this difference is profound. In metaphorical application, prior to 3/1/2>, a likeness between domains has been established (by 1/2/3>), and only then is the theory applied to the domain 'as if it was' something else. In foray, the usefulness of one theory for another domain is simply assumed, then tested. In metaphorical application, the domain is understood in terms of another conceptualisation. In foray, the theory determines another conceptualisation; minds are not only analysed as if they were machines, they are assumed to **be** machines. This also means that this interdisciplinarity works both ways – by representing the mind as a computer (representation is actually the name given by Richmond to 3/1/2>), it may also be possible to gain new insights about the mind.

Connectionist theories such as PDP employ theories of mind in this way. It is not the methods of analysis that are imported (at least not in this initiating vector), but rather, the domain of mind. It is not just that the same theories, or methods of analysis are applicable to the same domains. Rather, the import happens in 1ns, so this provides a kind of analysis, or representation, where mind and computer are perceived as being essentially the same.

2/3/1> - Another kind of computer

Initiated by the representation of mind and computers as essentially the same, another vectorial operation is set off: The domains are now parallel – this allows for the use of the same theories (3ns) in conceptualising the domain.

Because the cycle of foray was originally initiated by a theoretical likeness, the import of theory happening in this second vector is fairly straightforward. It takes as its starting point the method (2ns) formed in the preceding vector, that computers may be made to act more like the mind by applying theories of the mind to them. Through this cycle some features of theories of the brain are imported.

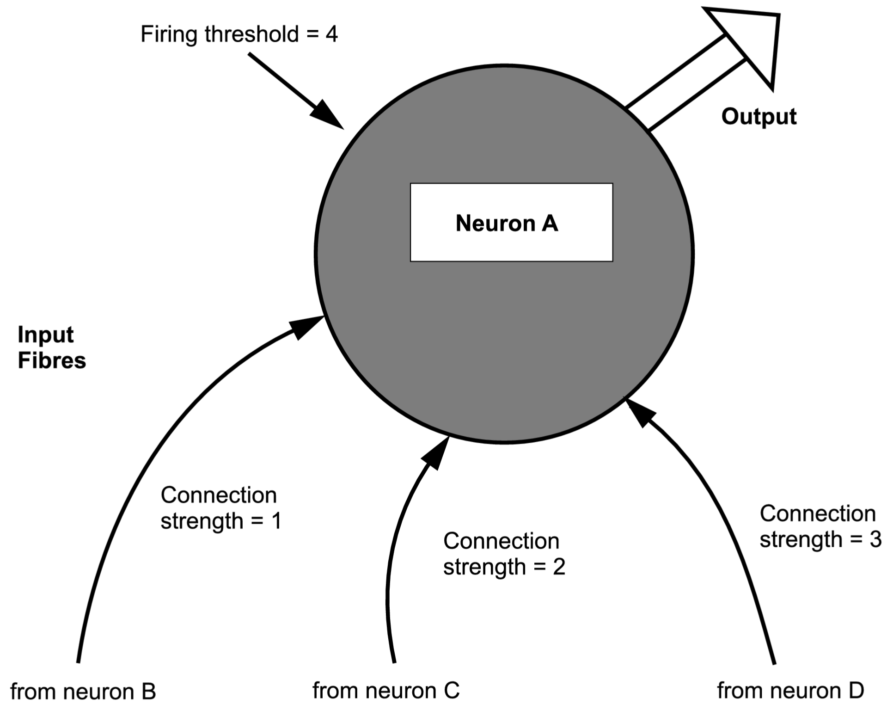


[De Koninck 2007, http://www.greenspine.ca/en/framed.html, Centre de recherché Université Laval Robert-Giffard]

The above figure shows a biological neural network. It is an image of a brain-slice obtained from a rat, and it clearly shows that the brain is a system of interconnected neurons. These may be studied from many different points of view. The above figure is taken out of the context of molecular and cellular neuroscience, the study of the genetics and chemistry of the brain. However, these theories on biochemistry are disregarded, and not imported into PDP, since they fall outside the scope of the theoretical unity established through the initiating vector. Only theoretical elements applicable to computers are imported, such as the idea that a neuron is "a receiver and transmitter of pulses of electricity(...)emit[ting] short bursts of electricity to its neighbours" [Copeland 1993, p. 183]. On one account of this view

> "all that matters about a neuron is whether it is firing or not (...) granted that the neuron is a simple On/Off switch, it can be viewed as a device for physically realising one of other of two symbols: Yes or No, True or False, 0 or 1.(...) McCulloch and Pitts explained how small groups of neurons could function as very simple symbol-manipulators. For example, three neurons can be connected together in such a way that the third fires when, and only when, both the other two do" [Copeland 1993, p. 185]

The account [McCulloch & Pitts 1943] interpreted by Copeland in the above is unrelated to computers (computers were at a very early state in 1943). But the theoretical unity of neuroscience and artificial intelligence is here taken to be the symbol systems hypothesis, so a theory that regards the brain as a symbol manipulating system lends itself to import into computing, giving a new view of how a computer might function: "The basic building blocks of a network are simple switch-like units that are either on of off. These are the artificial neurons. A network consists of a densely interconnected mass of these units."



[Copeland, p. 209]

The figure shows a simple artificial neuron. It has three input connections and one output. An impulse from neuron B is amplified by factor 1, one from neuron C by factor 2, and neuron D by 3. The 'firing threshold' of the neuron is 4, meaning that when the neuron is fired upon with more that 4, it will itself fire. Neurons of this kind may then be connected in networks.

In this way, the representation of the brain becomes a representation of a computer. The following citing clearly shows how elements in a computational neural network corresponds to elements of the brain:

> "the most common models take the neuron as the basic processing unit. Each such processing unit is characterized by an activity level (representing the state of polarization of a neuron), an output value

> (representing the firing rate of the neuron), a set of input connections, (representing synapses on the cell and its dendrite), a bias value (representing an internal resting level of the neuron), and a set of output connections (representing a neuron's axonal projections)." Rumelhardt et el. [1994]

Regardless of the obscurity (at least to me) of the neuroscientific terms, the citing illustrates that this vector results, not merely in a new way of looking at computers, but in a changed domain (ending in 1ns) altogether. This domain is one not only of computers, but one of the mind at the same time. This has happened through the import of theories (process in 3ns) from neuroscience. Which parts of these theories are imported is determined by a theoretical unity established in the preceding vector; Here, the brain and the computer are united through the symbol systems hypothesis; the theoretical elements that are imported from neuroscience are those occupied with symbol manipulation. This results in a product (in 1ns): A computer is now a system of interconnected nodes ('artificial neurons'), perceived as 'on-off-switches' that fire when brought above a certain threshold.

1/2/3> - Experimenting towards a theory

One final step is required in order to produce a proper theory (product in 3ns). At the point of the initiation of this vector, the network is merely an idea (1ns), it does not have any generative power (3ns), so that it may be used for developing actual implementations of PDP, or at least theories on how to design such machines. Such theories are developed through 1/2/3>, with their process in analysis (2ns), so this is empirical research; This final vector may be seen as gathering of evidence for the theory. The 'theory', of course, is still of a hypothetical nature, but it is strengthened by testing (by 'brute force' – the process is in 2ns) the idea (1ns) of the computer as a neural network. Through these tests, patterns and rules (3ns) may be derived, so 1/2/3>. As put by Rumelhardt et al. [1994]:

> “The strategy has been to develop simplified mathematical models of brain-like systems and then to study these models to understand how various computational problems can be solved by such devices. “

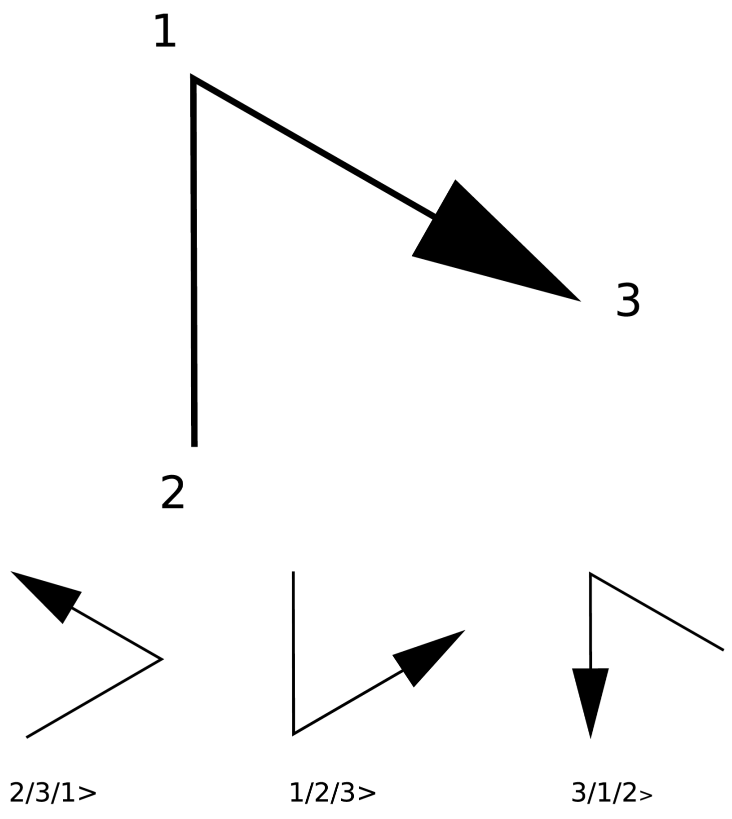
And they move on to explain how such studies are carried out:

> “The procedure is as follows: During training an input is put into the network and flows through the network generating a set of values on the output units. Then, the actual output is compared with the desired target, and a match is computed. If the output and target match, no change is made to the net. However, if the output differs from the target a change must be made to some of the connections.” [Rumelhardt et al., 1994]

So this is empirical science done from the outside-in. Some data are fed to the neural network, and the output is examined. If the output differs from 'the target' (the desired output), the inner workings, i.e. the connections, of the network are changed. Such changes, of course, are not done completely at random, but through mathematical models. These mathematical models are what I have called 'patterns and rules' for the design of neural nets. These models themselves are developed through the empirical study of the behaviour of neural networks. This is how the domain (1ns) of computational neural networks are studied empirically, in practice (2ns), in order to develop models, or theories (3ns) for the design of networks. So 1/2/3>.

While such behaviourist-dish approaches will often make use of foray, foray is not synonymous with behaviourism in any way. The theoretical 'likeness' initiating the cycle may also be hypothesised because one of the theories is of a general nature, e.g. theories of philosophy which, due to their generality, lend themselves to broad application. Foray may also be application of the theories in vogue. One needs only to go through my analyses above to see how ubiquitous metaphor theory has become. The same may be said for Activity Theory and a great number of other theories. In these cases, foray simply happens due to a scientist reading about a new theory somewhere, and then 'trying it out'.

Strengthening Extreme Programming – a case of refurbishment



The overarching vector in refurbishment is that of determination. It is reasoning from a likeness in analysis (2ns), i.e. shared practices, moving through the domain to develop new theory, or adjust an existing theory for use on a different domain.

Extreme Programming (XP) is a systems development methodology which is gaining considerable popularity. It is a modification of the waterfall, a software development methodology, which is often used in contemporary software development literature as the story to demonstrate how horrible software development used to be.

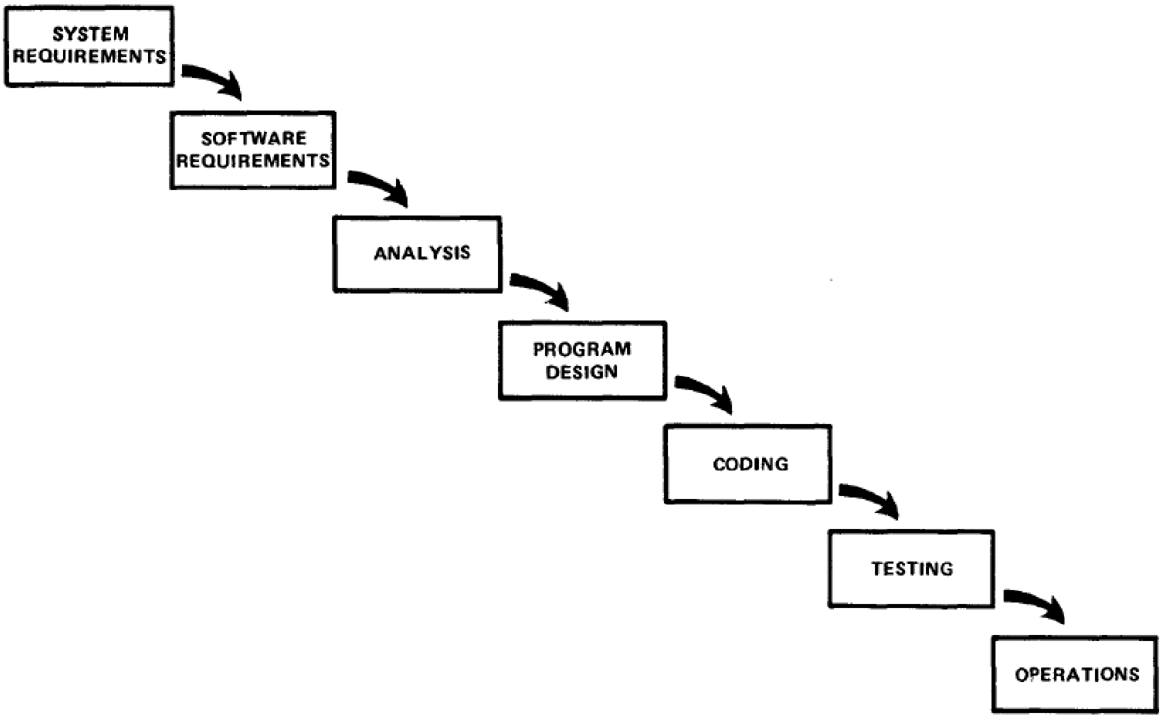


Figure 2. Implementation steps to develop a large computer program for delivery to a customer.

The figure above shows the waterfall, as originally presented in Royce [1970]. 'The waterfall model' as a label was given to the model later, but Royce remains the standard reference. The actual recommendation given by Royce [1970] was to adopt a more interactive approach to software development, and the intention of the above figure was to show an inadequate model. In spite of this fact, the model became US Department of Defence standard 2167, and subsequently spread into the software community [Larman 2004, p. 105].

XP, and other agile design methodologies, do not abandon the waterfall, rather, they acknowledge that the method, when put into practice (the cycle is initiated in 2ns), creates a number of problems, mainly stemming from the fact that the model puts a strong emphasis on planning in the early phases of the design process, while testing is conducted much later in the process. Working in this manner poses the obvious risk that errors and planning flaws are discovered too late [Larman 2004]. The solution to this problem is to transform the waterfall into an interactive method, i.e., instead of going through the waterfall once by planning and the coding, the project goes through a number of much shorter iterations of the model.

This methodology helps address the problems of the waterfall, but has problems of its own. Those problems mainly centre around a lack of planning (as opposed to too much planning in the waterfall), leading to projects with unstable objectives [Stephens 2003]. As such, much of the work that forms the theory of XP has to do with stabilising projects. Many of the problems are addressed through interdisciplinary manoeuvres: What forms the theory and methodology of XP is to a high degree a number of imports from other theories, that help address the problems of the practice of the XP methodology. The fact that these imports are initiated by variations in practice (i.e. the transformation of the waterfall methodology into an iterative method), and the import of theoretical notions (3ns) from other fields make XP a good example of what I call Refurbishment.

The literature on XP is largely atheoretical and intended for hands-on developers and business managers. Kent Beck, the creator and one of the main authorities on XP, however, makes the following academic-ish comments on the interdisciplinarity of XP in his 1999 *Embracing Change with Extreme Programming:*

> "The individual practices in XP are not by any means new. Many people have come to similar conclusions about the best way to deliver software in environments where requirements change violently. The strict split between business and technical decision making in XP comes from the work of the architect Christopher Alexander, in particular his work The Timeless Way of Building, where he says that the people who occupy a structure should (in conjunction with a building professional) be the ones to make the high-impact decisions about it. XP's rapid evolution of a plan in response to business or technical changes echoes the Scrum methodology and Ward Cunningham's Episodes pattern language. (...) XP's use of metaphors comes from George Lakoff and Mark Johnson's books, the latest of which is Philosophy in the Flesh. It also comes from Richard Coyne, who links metaphor with software development from the perspective of post-modern philosophy. Finally, XP's attitude toward the effects of office space on programmers comes from Jim Coplien, Tom DeMarco, and Tim Lister, who talk about the importance of the physical environment on programmers." [Beck 1999, p. 72]

That the initiation of the vectorial process of refurbishing the waterfall is in practice, is explicated in the above citing; While 'individual practices' that make up XP are not new, the juxtaposition of them is. Extreme Programming as laid out by Beck [2000] is quite comprehensive and comprises a lot of principles. For the purpose of showing the development of the method as a case of refurbishment, I will look into one of the principles. While it is probable, that the adoption of many of the principles could be analysed as vectorial movements, I have chosen metaphor, because it has a clear theoretical basis (In the citing above, Beck [1999] mentions *Philosophy in the Flesh* [1999] as a source of inspiration for the *metaphor* principle in XP). The main object of analysis, Beck [1999], falls somewhat outside of the academic realm, in that it is aimed at practitioners in programming and software development. I am analysing it into an academic context, and therefore I have chosen a principle that imports academic theory. Beck [1999] gives the following interpretation of the term metaphor: "The first thing we need is a metaphor, a shared story that we can turn to in times of stress or decision to help keep us doing the right thing." [Beck 2000, 30].

2/3/1> Using theory to address practical issues

A development project is hard to grasp as a whole. By the transformation of the waterfall into an agile process, this has become even more evident. So the cycle is initiated in 2ns, by problems concerning the practice of making sense of a development project as a whole. The practice (2ns) of software development is often complex and sometimes involves a lot of individuals. It is difficult for these individuals to keep overview of the project as a whole. A way of handling problems that are difficult to grasp cognitively may be found in theories about metaphorical reasoning: "we have subjective experiences of desire, affection, intimacy and achievement. Yet, rich as these experiences are, much of the way we conceptualise them, reason about them, and visualise them comes from other domains of experience" [Lakoff & Johnson 1999, 45]. The domains of desire and affection may have very little in common with systems design, but metaphor as a practice for understanding them and reasoning about them is quite similar. A large software system is, like the concept of desire, too rich and complex to reason about in its own terms, so it might be helpful to employ the practice of understanding it in terms of more basic knowledge. This unity of practice is the reason why the cycle of refurbishment is initiated in 2ns.

So the problem of something large and incomprehensive is addressed through importing theoretical elements from the theory on conceptual metaphor (3ns), namely the idea that something complex may be understood in terms of something familiar. This is a new view on the domain (1ns), which establishes the grounds for importing the more elaborate practices of metaphorical reasoning.

1/2/3> Adjusting metaphor theory for use in XP

This vector, in which the theory of 'conceptual metaphor in XP' is developed (it ends in 3ns), takes as its starting point the domain of systems development as something that now comprises the idea of metaphor. I will now look at how practices (2ns) of conceptual metaphor theory are brought in to the theoretical framework of XP. Wake [2000, pp. 88-89] states 4 purposes of the metaphor in XP: Common Vision, Shared Vocabulary, Generativity and Architecture. These four purposes correspond quite closely with some of the key concepts within metaphor theory.

Common Vision – Cultural metaphor:

> "To enable everyone to agree on how the system works. The metaphor suggests the key structure of how the problem and the solution are perceived. This can make it easier to understand what the system is, as well as what it could be." [Wake 2000, p. 87]

This has evident resonance with the concept of 'cultural metaphor':

The metaphor serves as a (micro)-cultural metaphor. In the same sense that "The most fundamental values in a culture will be coherent with the metaphorical structure of the most fundamental concepts in the culture" [Lakoff & Johnson 1980], the system metaphor in XP enables people in the macro-culture that is the development project to communicate about the project in terms that are equally meaningful to the developer, the manager and the customer.

Shared Vocabulary – Metaphor as linguistic phenomenon

> "The metaphor helps suggest a common system of names for objects and the relationships between them. This can become a jargon in the best sense: a powerful, specialized, shorthand vocabulary for experts. Naming something helps give you power over it." [ibid.]

The reason why naming something provides clarity, according to conceptual metaphor theory, is that metaphors, or naming, is not only words, they systems of understanding. In connection with the above notion of cultural metaphor, this provides a shared language in any culture – in this case specialised to a development team.

Generativity – Novel metaphors

> "The analogies of a metaphor can suggest new ideas about the system (both problems and solutions). For example, we'll look at the metaphor, "Customer service is an assembly line." This suggests that a problem is handed from group to group to be worked on, but it also raises the question, "What

> happens when the problem gets to the end of the line? Does it just fall off?" This can bring out important issues that might otherwise lurk and fester." [ibid. p. 88]

This is a re-interpretation of Novel Metaphor [Lakoff & Johnson 1999, pp. 66-67], i.e. the fact that cognitive structures can be extended: Entailments in in the metaphor become entailments in the software system as well.

Architecture - basic metaphor

> "The metaphor shapes the system, by identifying key objects and suggesting aspects of their interfaces. It supports both static and dynamic aspects of the system." [ibid.]

> "Architecture is just as important in XP projects as it is in any software project. Part of the architecture is captured by the system metaphor. If you have a good metaphor in place, everyone on the team can tell about how the system as a whole works." [Beck 2000, 90]

These ideas of the systems metaphor as governor of the system architecture corresponds to the notion that metaphors are used to reason with [Lakoff & Johnson 1999, p. 65]. Instead of reasoning within the framework of an immense and incomprehensible software system, the metaphor provides a simpler system to reason within.

All of the above are key analytical practices from conceptual metaphor theory that are imported in order to develop the theory of XP. Since this vector is initiated in the domain this is the viewpoint from which these practices are seen. It is a process in which they are brought in to the new context of systems development, and modified to suit that domain. This results in a preliminary, even hypothetical, theory, that has yet to be put into practice.

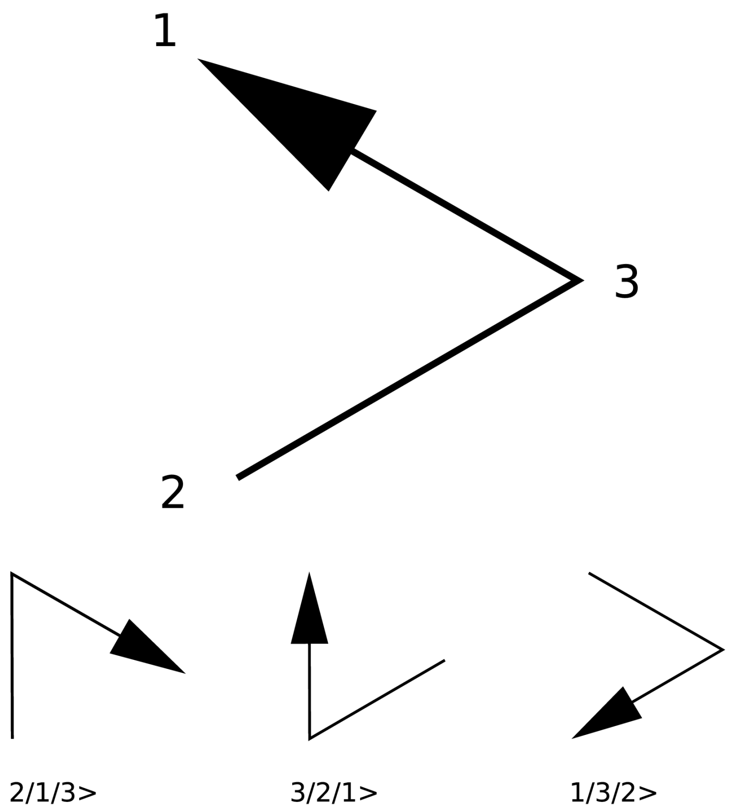
3/1/2> Metaphor as an XP practice

In this final cycle, the hypothetical theory of 'conceptual metaphor is XP' has to be transformed into a practice. From being imported chunks of theory (by 2/3/1>) and practices (by 1/2/3>), a practice is formed as part of XP as a whole. The literature on this transformation is sparse, since Beck [2000] devotes no more than 3 pages to it, despite the fact that he stresses its importance as being "The first thing we need". Wake [2000] gives the system metaphor a more thorough treatment, but it remains that the metaphor-principle of XP is often perceived as difficult to understand by XP adopters [Dubinsky & Hazzan, 2003, Lippert et al, 2003]

XP adopters have difficulties putting the systems metaphor into practice. If not entirely ignored, it is often implemented by means of ad-hoc solutions such as "Bulletin board with post it notes or story cards" [West 2002]. One large-scale quantitative study of the use of metaphor in XP development concludes that "There did not appear to be much use for any of the metaphors, for aiding in design, for communicating among the team, or for communicating with the customer." [Tomayko & Herbsleb 2003]

In other words, the adjustment of metaphor theory that happened in the previous cycle seems to create some tension with the domain of XP (1ns) from which the entire cycle sprung to begin with. The initiating vector 2/3/1>, imports theory (3ns) on the grounds of a similarity of practice (2ns) without much consideration of the difference of domains. Only this final vector has its process happening in the domain. The actual application of the (modified) theory does not happen until this final vectorial operation, and difficulties seem to arise in this phase. Perhaps this should not come as a surprise, since this vector, 3/1/2> corresponds to the abductive mode of reasoning. While abduction is considered by many to be crucial to scientific work, it does create knowledge that may or may not be true. The same holds for the cycle of adaptation: It results in a practice that may or may not be useful for the domain to which it is to be applied. When shopping for theories that might improve the practices of XP (2/3/1>) metaphor appears to be a good candidate. When tailoring metaphor theory to XP-needs (by 1/2/3>), the theory developed corresponds beautifully with the practices of conceptual metaphor theory . Yet, when the theory is put into practice, problems arise.

What has happened to ontology – a case of reconceptualisation



The overarching vector in reconceptualisation is aspiration. So this is reasoning based on a likeness in analytical methodology (2ns). Through theoretical reasoning (3ns), a new view, a reconceptualisation, of the domain (1ns) is derived.

Øhrstrøm et al. [2005] present a view on how Human-Centered Informatics is interdisciplinary in the paper *What happened to ontology*? Here, they suggest that computer science bears similarities to classic practices of creating abstractions of the world, formal ontology. That there is a resemblance between formal ontology and the object oriented perception of reality as classes and objects which inherits properties from more general classes, is self-evident. This application of ontology is, however, a special kind of 'application', if it makes sense to speak of application at all. As Øhrstrøm et al. note: “the import of the term ‘ontology’ from the realm of philosophy into computer science is not easily accounted for, and it would be wrong to assume that we here have a case of immediate transfer.” [p.434]. Despite the fact that the task has now been labelled 'not easy', I will make a simplified attempt at characterising this interdisciplinary move:

2/1/3> Encountering a likeness between practices

The main premise of *What has happened to Ontology* is that it is possible to identify a likeness between the practice of computer science and that of philosophical ontology. So Øhrstrøm et al. point out a likeness between practices (2ns): The paper makes the point that what goes on when knowledge is organised in computer science has a number of similarities to the task of organising knowledge in philosophical ontology. More specifically, they mention “multiple inheritance and the focus on classifying and structuring relations” as features common to the two practices. From this discovery, an inference is made, i.e., the resulting movement is initiated by a likeness of practice, so beginning in 2ns. This initiates a process in which some problems from philosophical ontology are identified, and it is argued that these problems (1ns) of epistemological nature may be imported into Computer Science, allowing for the development of theory, or at least of habits, in which the CS scientists take into account the problems imported from philosophical ontology. The inference then being, that when practices are similar, the domains are likely to be similar as well, or at least to present similar problems.

> “The notion of ontological commitment is an important aspect of philosophical ontology. However, this aspect is sometimes ignored in computer science, since ontology in these cases has been seen as mainly an information practice. Nevertheless, it turns out by closer inspection of the ontologies used in modern computer science that they do in fact presuppose some rather specific, but hidden ontological commitments.” [Øhrstrøm et al., p. 437]

Ontological commitment has to do with philosophical questions when working with ontology. These philosophical, or epistemological questions may be considered problems of the domain of ontology.(process in 1ns). Øhrstrøm et al. point to a number of different areas in which ontology workers might differ in position: plural or singular, Dependency of Domain and Perspective and position of product. The meaning of these areas are not of great importance here, what is important to note is that they are philosophical distinctions that are 'sometimes ignored' because ontology is treated as 'information practice'. So on this interpretation, Øhrstrøm et al. advocate that the problems posed (1nesses) by philosophical ontology may be fruitfully imported into computer science. The fact that awareness of ontological commitment has not made its way into computer science, is explained by the fact that within this discipline, ontology is regarded as 'information practice'. In other words, this is a difference in points of view (3ns); with regards to ontology, computer science is largely based upon a theory of 'no theory'. So the first thing to do if the problems from philosophical ontology is to be honoured in computer science is to incorporate these problems (1ns) into the 'point of view' of computer scientists, creating theory for ontology design (3ns). So this initial inference is 2/1/3>, the vector of determination. This preliminary theory, then, may lay the ground for reconsidering the domain of computer science.

3/2/1> A fresh look at Information Practice

In the preceding vector, the theory of ontology in computer science was enriched to also take into account problems from philosophical ontology (or, a theory based on philosophical ontology was created where there was no theory at all). In this next step, consequences of this theory for the domain of computer science are deduced, i.e. new problems are introduced into the domain (product in 1ns). This is done by reanalysing ontology design in CS while considering the problems imported. These problems are addressed by means of the methodologies (2ns) used to address these same problems in philosophical ontology, so this is import in 2ns.

As stated earlier, one element from the practice of philosophical ontology mentioned by Øhrstrøm et al. is that of ontological commitment: "the formulation of an ontology must involve statements about actual existence or non-existence of entities discussed in the theory." They attribute this to W.V.O Quine, an important scholar in philosophical ontology, and proceed to argue that

> "In general, it turns out that some kind of ontological commitment (in the philosophical sense) has to be involved in making an ontology even one made for practical purposes."

In the domain of computer science "ontology (...) is characterized by fragmented pieces of knowledge". When computer scientists design ontologies, they unconsciously see their domain in this way. One of the consequences of introducing practice from philosophical ontology into this domain is that these scientists are made aware that this is one way of seeing the domain, and that there are others (large, coherent ontologies making strong claims about the world as a whole). As such, a new problem has been introduced into the domain, a product in 1ns.

1/3/2> Refining the practice of Information Science

This refined view of Information Science calls for a modification of the practice of ontology design; The cycle of reconceptualisation ultimately ends in secondness. One aim of Øhrstrøm et al.'s paper is to make computer scientists aware of philosophical ontology (to explicate their ' ontological commitment'). Øhrstrøm et al. do not actually put theories of philosophical ontology into practice, but one of the works cited in the paper, Nicola Guarinos 1998 *Formal Ontology and Information Systems,* does so. The explicit focus of this paper is on the application side of formal ontology in information systems, a process quite clearly ending in 2ns: What he does is to import elements of philosophical ontology (so, import in 3ns) and put them into practice in information systems. He begins by an outline of the problem similar to the one in Øhrstrøm et al. [2005], arguing for awareness of theory from philosophical ontology on the part of computer scientists.

Interestingly, one of the way in which Guarino suggests this 'awareness', or view of the domain may have practical implications, pertains to the issue mentioned as an example in the preceding vector: That of large, coherent ontologies versus local ones designed for practical purposes (the problem labelled 'plural or singular' in Øhrstrøm et al., who also caution that this is not a dichotomy.). This is a much-discussed problem in philosophical ontology, but here, the problem is exactly that: philosophical. 'Singular' ontologies are seen “as each representing something comparably stable. In other words, each of ontologies should be understood as an intended attempt to describe how reality actually is.” [Øhrstrøm et al. 2005]. This is theory, and it may be understandable that it is often ignored in CS. However, from considering this problem, Guarino points out an important implication of this view. 'Plural', domain-dependent ontologies are developed for one application, be it a database, a user interface or something else, and are rarely usable for applications other then the one for which they were developed. Ontologies leaning more toward the 'singular' position, however, may be kept in an “ontology library” and reused for other applications. It may be noted that re-use of this kind is one of the core selling points of Object Orientation.

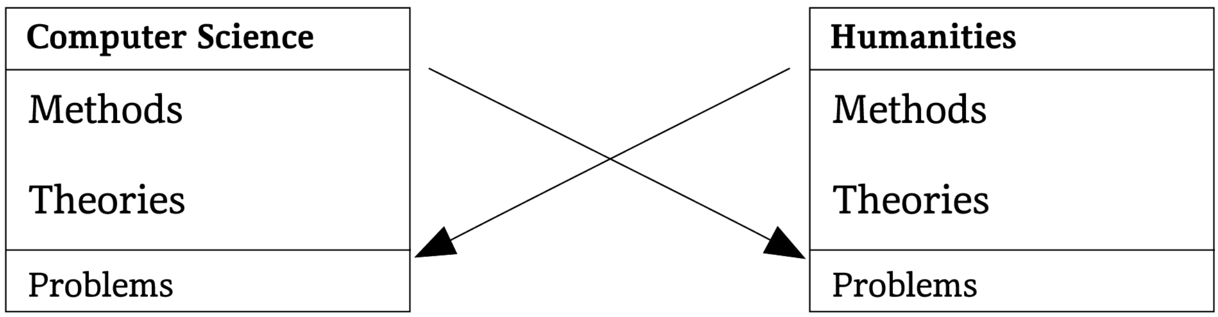
So this cycle as a whole results in a modification of the practice (2ns) from which the cycle was initiated – through a clockwise pattern of vectorial inference, the practice of organising knowledge in CS has been influenced by theory from philosophical ontology, a domain hitherto considered separate. The practices of ontology design in CS are enriched with problems from philosophical ontology. This allows computer scientists to see their domain in a new way. This has practical implications, one example being that distinguishing between plural and singular ontologies makes a difference for the re-usability of ontologies. Øhrstrøm et al. are right in stating that this is not a case of immediate transfer. It is a special kind of cross-discipline transfer that might be called “reconceptualisation”.

Conclusion

I have now identified six different modes of interdisciplinary transfers, and demonstrated that all these different modes can in fact be identified within the field of Human-Centered Informatics. This conclusion will explain why and how the model provides clarification to (1) what insights its application in the analysis has provided regarding the nature of Human-Centered Informatics, and (2) how interdisciplinarity unfolds within this scientific field. What are the practical consequences of the model? In explaining this, I will make a step away from the relatively close analysis of scientific texts done in the preceding chapter, and use the findings in my analysis as examples of how the model serves as clarification. I then proceed to discuss the suitability of Richmonds idea of vectorial movements for analysis of interdisciplinarity, and in closing I will reflect on how the model might find use in Human-Centered Informatics and related fields.

On the nature of Human-Centered Informatics

In order to illustrate what the model and the analysis have provided with regards to the nature of Human-Centered Informatics, I will briefly revisit Øhrstrøms interpretation of the field:



The examples given in the preceding chapter of six different ways in which interdisciplinarity may unfold within Human-Centered Informatics have made probable, that they the field is indeed methods and theories from Computer Science applied to problems traditionally belonging to the humanities and vice versa. However, it has also begun to demonstrate that this is one of several ways. Methods and theories are not simply 'applied'. In the process of importing methods and/or theories, the domains of the giving as well as the receiving science are taken into consideration as well. In 'Metaphor', theories on text lend themselves to application to the domain of systems development because the two domains share certain characteristics. Or new things may be seen in the domain due to 'Reconceptualisation' which allow theories and methods from other fields to apply. On an unreasonably naive interpretation, Øhrstrøms model may be seen as forcing or trying theories against domains other than that for which they were designed. And there are cases where this happens. 'Adaptation' is one, where a theory is more or less arbitrarily chosen (in the example it is less, since the choice rests on a traditional connection between psychology and HCI).

When looking through the analysis, it is evident that, at least in the examples I have chosen, that Computer Science and a number of (very) different aspects of the humanities have prominent status in Human-Centered Informatics. There have also been visits from other areas: Elements from Neuroscience and Operations Research, mainly influenced by economics, are examples.

How interdisciplinarity unfolds in Human-Centered informatics

Why the sequence of interdisciplinary import is important

Perhaps the most obvious objection to the model is that every interdisciplinary cycle imports elements from foreign theories in all three universes: domain, analysis and theory. The difference lies only in the sequence of these imports. However, this difference has profound consequences for the process of interdisciplinary import, and consequently for the knowledge produced.

One example of this is Richmonds [2006] interpretation of the vector of determination:

...a sign...

2/1/3> ..for an interpretant...

The object determines...

[p. 164]

The term 'determines' is key here (not surprisingly since it is the vector of determination). In the context of science, 2/1/3> is development of theory, since it ends in 3ns. Theory is a special kind of knowledge, it is a habit, that might be employed in mediating other signs and objects in the future. Just as importantly, it is initiated in secondness. Other than the fact that a similarity in secondness is what allows for interdisciplinary import in the first place, this has another consequence: Secondness is the determining vector. 2/1/3> in my analysis has been labelled 'refurbishment', and my case was that of Extreme Programming. This interdisciplinary movement was set off in practice, by something that can be observed, namely that developers had difficulties overseeing software development projects as wholes. This problems determined the choice of metaphor theory as part of XP practices (by the first vector, 2/3/1>), since metaphor theory addresses the exact problem of 'things that are to complex to perceive as wholes'. In this case, that which had been observed and encountered through practical use of existing theories of systems development (that is, in 2ns), determined the choice of metaphor theory as the perspective to adopt in the conceptualisation of the domain. This conceptualisation, in turn, had impact on the theory of XP.

Another example is that when Lytje expands the metaphor SOFTWARE IS A TEXT, It is the 'likenesses' identified in the domain that determine what parts of the theories may be imported into the theoretical realm of software. For example, theories on on onomatopoeia might not be importable into the analysis of the graphical layout of *Netscape Messenger.* The reason being, that while a likeness between software and texts has been identified, they do not agree in all respects. One difference between the layout of software (disregarding the labels of actual text that software often contains) is that a user interface, unlike a text, cannot be read out loud. So applying theories on onomatopoeia, or general phonetics, for that matter, seems implausible. In this case, the cycle began from a new conceptualisation (1ns) of the domain. This conceptualisation determined the choice of theory (the vector was 1/3/2>, so the first movement was from 1ns to thirdness) as theories that are traditionally applied to texts.

Which realm acts as determining depends on where a similarity is encountered or established.

Does similarity appear out of nothing?

As some of the examples of the different modes of disciplinarity have already suggested, similarity does not come out of nothing.

"The forms of the words similarity and dissimilarity suggest that one is the negative of the other, which is absurd, since everything is both similar and dissimilar to everything else." [CP 3.567]

So, in this fragment from Peirce, all things might be similar. On this note, similarity is not something inherent in things, but rather dependent on perspective, or at least something that is established, not pre-existing. Or it might be existing similarity that is discovered by some process. The analysis I have made employing the model begins to demonstrate this, in that similarity may be established, or discovered, in either the domain, by practical analysis or in theory. When looking at the analysis, it may be taken to suggest that even within these different realms, similarity may be established in different ways. Examples of words that describe the ways in which this may happen are chance, idea, force, necessity and variation. This list of words is not exhaustive, and the model is not itself helpful in explaining the difference between different kinds of similarity. Within the scope of the model, it does not make a difference directly. In order to explain how the model has practical relevance, I will give some examples of how interdisciplinary cycles may be driven by the three different realms, and of how the similarity may come into being. It should be noted that the way in which a similarity comes into being does not necessarily initiate the exact vector it does in the examples.

How interdisciplinarity is driven

A should be clear at this point, cycles may be driven by the domain, by analysis or by theory. In the following, I will give a number of examples of situations that might give rise to a cycle driven by the different realms.

Domain-driven cycles

These cycles are driven by the domain, by some kind of likeness. The likeness may arise through changes in the domain, or it may be discovered.

Emergence of new technologies

The emergence of the computer is one example of a new phenomenon altogether. It may not be possible to study such phenomena within the established sciences. This kind of initiation not only holds for technologies, but for illnesses, or a new type of plant. Anything that has inherent qualities that cannot be realised through the use of analytical methods and theories from within the science to which they present themselves. Such cases may also be interpreted as 'chance variations' in the domain, new possibilities in firstness that calls for new techniques in order to become realised. In some cases, they may give rise to new sciences that cross the boundaries of traditional ones. The field of Human-Centered Informatics is a prime example of this. More specifically, analogy, exemplified by the case of Object Oriented Analysis, is such a response to the emergence of the computer, i.e. a variation in the domain. The actual variation in the domain in this case (as well as in the case of the entire field of Human-Centered Informatics) the interdisciplinary movement is initiated, not as much by the computer in itself, but by the problems that the computer has introduced. As it was at the time of the conception of OOA, Computer Science dominated the theoretical scene in Systems Development. However, as described in the analysis, problems with the workers using the computers arose, that could not be addressed within the field of Computer Science, so a wish emerged to develop theories that might address these problems, illustrated by the vector of analogy 1/2/3>, ending in 3ns. The first vector in the cycle of analogy is 1/3/2>, and this shows clearly how problems in the domain determine the choice of theory. In this sense, it is problem-oriented research, since the traditional affiliation of the theories chosen in principle do not matter, as long as they are able to address the problems at hand and provide useful analyses. In the case of OOA, Nygaard [1992] saw a likeness between the problems at hand and that in Operations Research, which then became one of the prominent theories in the development of OOA.

Sudden discovery of a likeness

The domain is firstness, i.e. it is possibilities, airy nothingness, that may come into being. This realisation may take the form of a discovery of a likeness between domains. We saw a good example of this in the metaphor cycle. Lytje **came to realise** that books and software has inherent qualities that are similar enough to analyse them as the same thing. Similarly to what happened in OOA, these similarities between books and software then drive the choice of theory and conceptualisation of the domain. But whereas analogy, the cycle exemplified by OOA immediately imports theoretical elements from OR in its first vector, the cycle of metaphor does not import theory until the final vector. instead, it develops a hypothesis (uncertain theory) by 1/2/3>, the consequences of which is then derived in the following vectors. This means that the import of theory from textual analysis is determined by a pre-developed (by 3/1/2>) analysis, making it quite clear which elements of the theory should be imported in the final vector. As such, because the likeness is established from the domain by forming a hypothesis (1/2/3>), which is then qualified against the domain (3/1/2>), and only then is foreign theory applied to derive its consequences (2/3/1>), the degree of assumption in this cycle is quite low. This is further amplified by the fact that Richmond interprets the vector 1/3/2> as peircean idealised knowledge production:

“abduction (the case is possible)

1/3/2|> deduction (the result is necessary)

induction (the rule is probable)” Richmond [2005]

Analysis-driven cycles

These cycles are driven by 2ns in the sense that they are initiated by practical problems or by something that is encountered or uncovered through analysis of some kind. This 'analysis' might simply be 'a certain way of doing things' as was the case with systems development using the 'waterfall' in the example of Extreme Programming.

Practical development

The term practical development may invoke associations towards the professional world of software development rather that academia. And indeed, the literature in the example of Extreme Programming, Beck [1999] is aimed at practical systems development. Yet, as established in the analysis, he draws upon theories and methods of an academic nature in the development of the 'metaphor-principle' in XP. As such, this is an interesting example of how academic theory may play a role in professional practice in an unexpected way.

Practical in the example of XP does not mean atheoretical. No hypothesis formation (until the very last vector which has the character of hypothesis – and fails). The initiating vector moves directly from practice to a conceptualisation of the domain, importing metaphor theory assuming that it will be helpful, by 2/3/1>. The most interesting finding in the example of 'metaphor in XP' is that it seems to fail in practice. The vector might provide an explanation to why the application of metaphor does not fit the domain of XP. This is an early import of theory, without consideration for the domain, and what is more, it is determined by observations in practice. The cycle then immediately proceeds to import methodology from metaphor (1/2/3>), developing the actual metaphor-principle, but still without any epistemological import, i.e. comparison of the two different domains. This happens only in the final vector, where the principle is to be put into practice. In XP, this final operation seems to fail in many cases.

Methodological Unison

One of the dimensions on which the division of sciences is often argued for is that of methodology. Different sciences have different ways of doing science, and different realisations of the domain. However, Faye [2000] argues for a methodological unity of science, i.e. despite the theoretical differences and differences in the domains, all sciences essentially share one method. This is a radical and somewhat provocative view. However, it may be moderated into a version in which some sciences can share methodological characteristics despite theoretical differences and different objects of research. This is probably part of the reason why Human-Centered Informatics has been successful in applying theory from the Natural Sciences to the humanities. On a slightly more specific note, two sciences may display similarities in their analytical methodologies, There are cases (one of these being ontology in computer science, as shown in my example) where sciences with altogether different objects of inquiry approach them in a similar manner. Similarly to what happened in the example *Software as Text, What has happened to Ontology* rests on a discovery of a likeness between philosophical ontology and modern computer science, only in this case the likeness is methodological. And like in *Software as text* the first vector establishes the likeness as a hypothesis (preliminary theory, by 2/1/3>). This is contrary to XP where the initial likeness was assumed to hold, since it moved directly from likeness to conceptualising the domain. Not so in reconceptualisation (the cycle followed by *What has happened to Ontology)*, where the next vector imports the methods of Philosophical Ontology to give a fresh look at Information Practices, by 3/2/1>. With this conceptualisation in place, the ground is laid for selection of theoretical elements from Philosophical Ontology that might result in enriching the Computer Science with elements from Philosophical Ontology. Contrary to XP, this import of theory is preceded by a reconceptualisation which allows for an informed choice of theoretical elements for import.

Theory-driven cycles

Discovery of new theories by chance

The interests of a scientist is seldom limited to his own area of study. Even if someone tried, it would be impossible not to come across theories from other fields, be it at conferences, in casual conversation or by watching TV. The example of Foray in my analysis, PDP, might not be serendipitous to such a high degree. Nevertheless, no-one can say for certain to what degree the computer and the brain share characteristics. If this was the motivation, the initiation would be by domain rather than theory. But this was not the case. In the analysis above, I interpret PDP as being born from Artificial Intelligence, the dream of making computers do the same thing as brains. The thesis was that this could be achieved by applying theories of the brain to the computer. In actuality, there need not actually be much 'likeness' between the theories traditionally employed and the new ones discovered. In PDP, the 'likeness' rested on a hypothesis, roughly being that the human brain and the computers share so many characteristics that they may benefit from the same theories. But it need not even be so: Interdisciplinarity in this way might happen because of a scientist finding a theory interesting and wanting to try it. Among the examples of initiation of this kind is also scientists who have developed some theory for one domain, which they then proceed to apply to others. One example is the metaphor theories of Lakoff and Johnson which started out as a linguistic theory [Lakoff & Johnson, 1980], later became a philosophical one [1999], and in the meantime was applied to every living thing [Lakoff 1987].

Response to scientific, philosophical or societal movements

These are general scientific trends. Trends here taken in all its meanings. It may be everything from some broad philosophical and/or societal movement (post-structuralism and such) to this years scientific fashion. Like most cases probably do, the application of Activity Theory used as example above falls between these extremities. This theoretical direction won some popularity in psychology, and because Human-Computer Interaction and psychology in many respects share their theoretical foundation, it might have seemed appropriate as well as potentially fruitful to relate to this general tendency and assume that when it seems to be of value in psychology, it may well be so in Human-Computer Interaction.

If I tend towards stultifying this kind of interdisciplinarity in parts of the above, I am not alone in doing so:

"Denne slags imperiale mestervidenskaber [er i en vis forstand] tværvidenskab og forsøger ofte at legitimere sig som sådan: i deres imperiebyggeri forestilles videnskabernes grænser at bortvitres – man ganske vist kun fordi én reduktive sandhed så antages at brede sig ud over hele feltet. Det er naturligvis ikke den slags tværvidenskab, der har været holdt skåltaler til i det seneste tiår"

"This kind of imperial master-sciences [are in some sense] cross-science and often tries to legitimise themselves as such: in their empire-building the boundaries of science are imagined to crumble – although only because one reductive truth is taken to spread across the entire ground. This is of course not the kind of cross-disciplinarity that toasts has been proposed to in the latest decade" [Engelhardt & Stjernfeld 2007, p. 11]

As may be read from the above, interdisciplinarity of this kind is sometimes spoken about in a derogatory manner. And perhaps with justification when talking about 'imperial' sciences as in the above. However, reasonings of this kind play a role in science. Keep in mind that the abductive form of reasoning is motivated in thirdness, and is the only form of reasoning capable of producing genuinely new knowledge. This is seen in Adaptation, the case of Activity Theory in interaction design, where the choice of Activity Theory determines the new way in which the domain of Interaction Design is seen; In other words, the consequences of mediating the domain with analyses using Activity Theory are deduced. And it is hard to deny that that Nardi & Kaptelinen [2006] succeeds in establishing Activity Theory as a useful perspective for HCI, but it may also be said that this is due to pure luck, since the first vector of the cycle of adaptation, 3/2/1>, proceeds directly to applying the methods and techniques from Activity Theory, giving a conceptualisation of HCI that determines the remainder of the cycle. Of course, the fact that the theory comes from a related field traditionally related to HCI, psychology, somewhat alleviates the danger in trying out a new theory against a domain other than that for which it was designed, but adaptation is still risky business.

Application of theory by decision

Sometimes theories for application may be chosen more or less arbitrarily. In these cases theory is chosen due to personal belief and preference. Sometimes this is even explicated in titles such as 'A Habermasian approach to organizational change', where a certain theoretical standpoint is established from the beginning, without this standpoint being determined by the domain nor by methods of analysis.

Such arbitrary applications are open to criticism for being unscientific, for being controlled by the theory etc., But application of this kind are also creative; it is one way in which genuinely new knowledge may be created, since it provides a fresh look at some domain. The application of Activity

Theory to HCI is one such example, although in this case, the 'decision' to employ Activity Theory was due to tradition.

In special cases, theory-driven interdisciplinarity does not even emerge out of decision, but rather due to 'brute force' in a more everyday sense than its peircean one. In the report FORSK2015, passed by the Danish government as a strategy for research over 7 years, interdisciplinarity is a major thread. The theme of research labelled “Intelligente Samfundsløsninger”, “Intelligent societal solutions” is one example that influences Human-Centered Informatics:

“There are great potentials in thinking together research and development in ICT with a number of substantial societal areas of practice. The focus for a strategic research effort is systems integration and development, in which new, innovative ICT-based solutions within e.g. heath and environment are developed in close, cross-disciplinary cooperation with other areas of science. A research effort within intelligent societal solutions must raise the quality in public production and contribute to better solutions for a number of areas of society. The research must be cross-disciplinary with a high degree of knowledge exchange and cooperation between researchers in ICT and users and researchers from other areas of practice”

[Forsknings- og Innovationsstyrelsen 2008]

Here, interdisciplinarity is not something that is determined by the domain nor the problems in question. It is something the researchers must do in order to provide good ICT solutions to problems in society (and to receive funding). This is a special kind of interdisciplinarity indeed; the report emphasises healthcare and environmental research as areas that must work with researchers in ICT. The representatives of these areas of research are then forced to arrive at some kind of similarities in their theoretical bases. As should have been established at this point, this 'forced similarity' then becomes determining for the integration of the disciplines involved in the research. Since the aim of research, according to the report is to “develop applications that have been fused with the processes and uses that the solutions are to function in” [ibid.], i.e. to develop practical solutions, 2ns, in vectorial terms this would probably be interdisciplinarity by 3/1/2>. In the example of adaptation, I stated that this kind of interdisciplinarity was a success in that specific case, but that it need not have been, since the domain is considered only in the very last vector.

A tangible outcome of this exact kind of research is the Electronic Patient Journal (EPJ), a system for digitalising patient journals in the Danish health care system. It has received much criticism, one of the points being that: “According to it, the doctor always works ideally, i.e. structured. The diagnose is made, the treatment is immediately planned, objectives for the outcome of the treatment are defined, and the actual result is compared with this objective” [Toft 2003]

It seems clear that a system of the kind described is the product of flow-diagrams and simulation-like models of the task the system should support. In other words, the research that has gone before the development of this system has clearly been determined by the theoretical framework of systems development. As such, the EPJ is an example of 3/1/2>, adaptation. Even the word adaptation seems appropriate here, since it is systems development adapted for healthcare. The problem with interdisciplinarity of this kind is the same as in refurbishment, where we saw metaphor in Extreme Programming fail in practice, the reason being that the actual domain for application was not considered until the very last vector. This characteristic is shared by adaptation: The actual domain of medicine is only considered in the final vector, which is 2/1/3>, a move from analysis to theory by considering practice – so this would probably be the evaluation phase. Instead of looking at the domain, the first vector in adaptation is 3/2/1>, a move from theories of systems development (3ns), looking at the work practices of healthcare (2ns) in order to arrive at a conceptualisation of it (1ns). So the while the ideal practice of healthcare has been 'imported' into the systems development process, the actual domain (1ns) has not been taken into consideration. As it turns out, the object (1ns) of the physicians work, i.e. the human body, is too complex to be fitted into the EPJ. Due to the nature of 3/1/2>, adaptation, this is discovered late in the process. The vectorial model of interdisciplinarity might devise different approaches to developing the system in which the domain is determining, such as 1/3/2>, metaphor.

Of course, the difference lies not only in which vector is the first in a given cycle. For the following cycles, the product of the preceding cycle acts as initiation, and thus as a determining factor. So while the difference between the vectors lies 'only' in the sequence of import and the line of reasoning in the triad, this difference is important indeed for the nature of the import as well as the knowledge produced.

Suitability of the model

Richmonds theory on |>*, and as such the model of interdisciplinarity, may on the whole be described as somewhat rigid, because of the relatively high degree of formality imposed by the identification of exactly six different vectors and their diagrammatic representation. In this rigidity lies many of the strengths of the model, but at the same time |>* has its drawbacks and weaknesses, many of which also pertain to its rigidity.

Rigidity as a weakness

The obvious drawback of the rigidity of the model is evident from my analysis. When applying the model, it controls the structure of the analysis, causing it to be quite theory-driven, with the obvious risk that the theory determines the conclusions. When this might seem to be the case in my analysis, this is because it is not intended to be an analysis of the field as much as it is to demonstrate that the six cycles can in fact be identified within the field. If the model were to be applied to more comprehensive analyses of Human-Centered Informatics it would be desirable to let the the texts determine the vectors instead of the other way around.

Another critique might be that the model is very general and requires a high level of abstraction. This issue is inherited from Richmonds theory, which in turn inherits it from the semiotics of Peirce. As I will argue, however, Richmonds diagrammatic representations of semiosis as vectors is one of the clearer explanations of Peirces idea.

It may also present a problems, that in its present form, the model does not allow for cycles that “change direction”, i.e. do not follow the path laid out by the six vectors. This however, might be symptomatic that it is being applied at too high a level.

Rigidity as a strength

In an interesting paper on interdisciplinary knowledge production, Burger & Kamber [2003] provide the following definition of interdisciplinarity as a cognitive task:

“The basic cognitive goals of transdisciplinary (as discriminated from disciplinary) research can now be defined as producing knowledge under systematic methodological restrictions that will describe the complex subject matters of transdisciplinary problems more comprehensively” [Burger & Kamber 2003]

What the application of |>* to interdisciplinarity represents is exactly a move away from attempting to frame interdisciplinarity in linguistic terms. “Systematic methodological restrictions” is a most accurate description of what |>* provides. It certainly describes the 'complex subject matters' more systematically. The fact that the model dictates the structure of analysis is a strength in this sense:Because any interdisciplinary movement can be categorised under six vectors, it makes it easier to point towards differences and similarities with different situations of interdisciplinary import. Whether the model is comprehensive is a matter for debate. The graphical representations are very clear, but the underlying body of theory is not exactly simple.

Peirces idea of semiosis may be difficult to comprehend, but |>*is a quite clear representation of it, partly due to the way in which vectors break down into sub-vectors, partly in that it supports diagrammatic reasoning, emphasised by

Peirce as “the only really fertile reasoning” [CP 4.571] in connection with his likewise diagrammatic existential graphs. Even when it is applied only at one level, it shows clearly how scientific reasoning and cross-disciplinarity tie together. As a consequence of the very abstract nature of the model, it applies to very broad fields, such as Human-Centered Informatics.

At the outset of this report, I defined one of the problems of interdisciplinarity as: How does new knowledge arise from existing knowledge from more than one discipline?. This 'how' covers a vast variety problems, one of these being the question of how it is determined which elements are selected from a present body of knowledge and which are not. In the context of interdisciplinary science, this is the problem of which elements are chosen for transfer from one science into another and which are not. The model seems to suggest an explanation: A likeness, or a preceding vector determines how elements are chosen for import.

Overall the model, although it may be insufficiently fleshed out, shows promising potential for analyses of how Human-Centered Informatics is interdisciplinary.

Applicability

I will now make a step even further away from the field of Human-Centered Informatics to reflect on its potential for more general application to related fields (and possibly even unrelated ones).

As an analytical method, the model is a framework for analysing interdisciplinary scientific fields. In the case of Human-Centered Informatics, it was relatively easy to identify every one of the six vectorial movements. Perhaps not surprisingly, since Human-Centered Informatics was chosen because it is a good example of a relatively mature, inherently interdisciplinary field. As such, over time it has imported a large body of problems, methods and theory from other fields through complex and intertwined operations, to which the model can provide clarification. In other fields, it may be more difficult or even impossible to identify all the vectors. As an example, fields arising around some new object of research, would at the outset probably have overweight of cycles beginning from that object, i.e. from the domain, in determining the theories and methods that apply. Only later in the process is a sufficient number of theories developed (or 'habits taken on') that interdisciplinary operations can originate from theoretical similarity. Likewise a field that takes a new (theoretical) look at some well known object would be probable to have an overweight of cycles set off by that theory. An example would be 'centre for applied experience economy', one of the more newly established research centres at Aalborg University, which is highly interdisciplinary, but obviously inspired by the idea of 'The Experience Economy' [Pine & Gilmore, 1999]. At the same time, the patterns formed by vectorial patterns could differ substantially between sciences belonging to different traditions, e.g. vectors initiated by observation might have a more prominent role in the natural sciences. In this way, vectorial analysis can produce knowledge and awareness about the maturity of a given scientific field a as well as its nature. This naturally leads to the question: Can the model be used generatively, to develop scientific fields?

Richmonds own application of |>* actually has such a generative focus; He applies |>* to virtual communities engaged in collaboratory work, as a method for increasing shared understanding. Interdisciplinary science is, by definition, some form of collaborative work, although the collaboration may only consist in reading each others papers. This core idea of |>* also applies to the model of interdisciplinarity. Through a special kind of diagrammatic reasoning, diagram manipulation, the model supports awareness of the nature of interdisciplinary import. As such it may be helpful in developing scientific areas, by allowing scientists to see similarities to other disciplines which they may not have seen otherwise, and to be aware of the lines of reasoning, and the difference in nature of the knowledge produced when working interdisciplinarily in different ways. If strong patterns form in the analysis of a science towards one type of vector or the other, it may something fruitful to reflect upon. It might be because of the nature of a field: As already mentioned, natural sciences may show patterns different from those in sciences within the humanities. Or it may be that the field is a design-science in which a large part of the work is 'practical', happening in 2ns. As such, it might be possible to influence the nature of a science through vectorial design. The model can also find use in cultivating interdisciplinarity; Administrative bodies put pressure on scientists to work interdisciplinarily, also on sciences that might not have a tradition of such research. In these cases, the model can provide guidelines and a systematic methodology of different ways in which ideas from different disciplines may interact.

It is my view that this diagrammatic methodology for developing interdisciplinary scientific fields is clearer and more operational than a complex of definitions such as trans-, cross- and interdisciplinarity, problem-orientation, interdisciplinary competence and a horde of other more or less ill-defined concepts that often surround interdisciplinary research.

References

Bechtel, W. & Graham, G. (eds.). 1999. *A Companion to Cognitive Science*. Oxford: Blackwell.

Beck, K. *Extreme programming explained: embrace change.* Boston, MA : Addison-Wesley Longman Publishing Co.

Billman, D. 1999. Representations. In: Bechtel, W. & Graham, G. *A companion to Cognitive Science*. Oxford: Blackwell.

Burger, P. & Kamber, R. 2003. Cognitive Intergration in Transdisciplinary Science. - Knowledge as a key notion. In: Issues in Integrative Studies. No. 21, pp. 43-73.

Bødker, S. 1989. A Human Activity Approach to User Interfaces. In: *Human-Computer Interaction*. Volume 4, pp. 171-195. Lawrence Erlbaum Associates, Inc.

Bødker, S. 1991. Activity theory as a challenge to systems design. In H-E. Nissen, H. K. Klein and R. Hirscheim (eds.): *Information Systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier, Amsterdam, pp. 551-564.

Carroll, J.M. 2003. HCI Models, *Theories, and Frameworks: Toward a Multidisciplinary Science.* Morgan Kaufmann.

Churchman, C.W. et al. 1957. *Introduction to Operations Research*. New York: Wiley.

Clausen, H. (ed.). 2007. *Human Centered Informatics Doctoral Research School*. http://www.hci.hum.aau.dk/about/index.html

Copeland, J. 1993. *Artificial Intelligence*. Oxford: Blackwell.

Dahl, O. & Nygaard, K. *How Object-Oriented Programming Started*. http://heim.ifi.uio.no/~kristen/FORSKNINGSDOK_MAPPE/F_OO_start.html

Dubinsky, Y. & Hazzan, O. 2003. Using Metaphors in eXtreme Programming Projects. M. Marchesi and G. Succi (Eds.): *XP 2003* LNCS 2675, p. 1016. Berlin: Springer.

Engelhardt, R. & Stjernfelt, F. 2007. Indledning. In: Engelhardt et al. (eds.). *Tankestreger – tværvidenskabelige nybrud.* Copenhagen 2007: Danmarks Pædagogiske Universitetsforlag.

Fauconnier, G. & Turner, M. 2002. *The Way We Think*. New York: Basic Books.

Feye, J. 2000. *Athenes Kammer – En filosofisk indføring i videnskabernes enhed*. København: Høst.

Fillmore, C.J. 1982. Frame Semantics. The Linguistic Society of Korea, *Linguistics in the Morning Calm*, selected papers from SICOL-1981. pp. 111-137.

Føllesdal, D. et al. 1990. *Politikens bog om Moderne Videnskabsteori*. København: Politikens forlag.

Graaff, E. & Kolmos, A. 2007. *Management of Change: Implementation of Problem Based and Project Based Learning in Engineering.* Sense Publisher.

Guarino, N. 1998. Formal Ontology in Information Systems. In: Guarino, N. (ed) *Proceedings of FOIS'98,* IOS Press, Amsterdam: Trento, Italy, pp. 3–15.

Hasle, P. 2003. "Software som Tekst". En fortolkende anmeldelse. In. Schärfe, H. (ed.) *Impact*. Aalborg Universitetsforlag.

Hove, J. & Schärfe, H. Blended Icons. / Hove, Janus ; Schärfe, Henrik. I: *The 26th International Conference Of The Poetics And Linguistics Association : The State Of Stylistics*. University of Joensuu - Finland, 2006. 1 s.

Hove, J. 2007. *The interplay between Cognitive Science and HCI – a preliminary analysis*. Semester paper submitted to Aalborg University.

Kaptelinin, V. & Nardi, B. 2006. *Acting with Technology*. Cambridge: MIT Press.

Klein, Julie. 1990. *Interdisciplinarity: History, Theory, and Practice*. Wayne State University Press.

Lakoff, G. & Johnson, M. 1980. *Metaphors We Live By*. Chicago: University of Chicago Press.

Lakoff, G. & Johnson, M. 1999. *Philosophy in the Flesh. The Embodied Mind and its Challenge to Westerm Thought*. New York: Basic Books.

Lakoff, J. 1987. *Women, Fire and Dangerous Things.* Chicago: Chicago University Press.

Larman, C. 2004. *Agile and Iterative Development. A Manager's Guide.* Boston: Pearson Education.

Leontjev, A.N. 1978. Activity, Consciousness, and Personality. http://www.marxists.org/archive/leontev/works/1978/index.htm

Lippert, M. et al. 2003. Metaphor Design Spaces. M. Marchesi and G. Succi (Eds.): *XP 2003*, LNCS 2675, pp. 33–40. Berlin: Springer.

Lytje, I. *Software som tekst*. 2000. Aalborg: Aalborg Universitetsforlag.

Mathiassen, L. et al. 2001. *Objekt Orienteret Analyse & Design*. Aalborg: Marko.

Mautner, T. (ed.). 2000. *The Penguin Dictionary of Philosophy*. London: Penguin.

McCulloch, W.S. & Pitts, W. 1943. A logical Calculus of Ideas Immanent in Nervous Activity. *Bulletin of Matemathical Biophysics*, 5, pp. 115-133.

Melmed, Raphael N. (2001). *Mind, Body and Medicine: An Integrative Text.* Oxford University Press Inc, USA, 191-192.

Merleau-Ponty, M. 1962. *Phenomenology of Perception.* New York: Humanities Press.

Newell, A. & Simon, H.A. 1976. Computer Science as empirical enquiry: Symbols and Search. In: Haugeland, J. (ed.) *Mind Design: Philosophy, Psychology, Artificial Intelligence.* Cambridge: MIT Press.

Nissani, Moti, 1995, Fruits, Salads, and Smoothies: A Working Definition of Interdisciplinarity. In *Journal of Educational Thought/Revue de la Pensee Educative*, v29 n2 p121-28 Aug 1995.

Nygaard, K., 1992. How Many Choices Do We Make? How Many Are Difficult?, pp. 52-59 in *Software Development and Reality Construction*, Floyd, C., ZŸllighoven, H., Budde, R., and Keil-Slawik, R., editors. Springer-Verlag, Berlin 1992.

Paahus, M. 1995. Hermeneutik. In: Collin, F. & Køppe, S. *Humanistisk Videnskabsteori*. København: DR Multimedie.

Peirce, C.P. 1958-1966. *Collected papers*. Edited by Charles Hartshorne and Paul Weiss. Vols. 7-8 edited by A.W. Burks. Cambridge: Belknap Press of Harvard University Press..

Peirce, C.S. 1992, 1998. *The Essential Peirce*, 2 vols. Edited by Nathan Houser, Christian Kloesel, and the Peirce Edition Project. Bloomington, IN: Indiana University Press.

Pine, J. and Gilmore, J. 1999. *The Experience Economy*, Harvard Business School Press, Boston.

Richmond, G. 2005. Outline of trikonic |>*k: Diagrammatic Trichotomic. In: *Conceptual Structures: Common Semantics for Sharing Knowledge*. LNAI, Volume 3596/2005. Berlin: Springer. pp. 453-466.

Richmond, G. 2006. Trikonic Analysis-synthesis and Critical Common Sense on the Web. In: Hitzler, P. et al (eds.). *Inspiration and Application*. ICCS 2006. pp. 158-171.

Rosch, E.H. 1973. Natural categories. In: *Cognitive Psychology* 4, 328-350.

Royce, Winston (1970), Managing the Development of Large Software Systems, *Proceedings of IEEE WESCON 26* (August): 1-9.

Rumelhart, D.E. et al. 1994. The Basic Ideas in Neural Networks. In: *Communications of the ACM,* March 1994, vol. 37, nr. 3. pp. 87-92.

Schärfe, H. 2003. Forord. In. Schärfe, H. (ed.) *Impact*. Aalborg Universitetsforlag.

Sowa, J. 2000. *Knowledge Representation*. Brooks/Cole.

Stephens, M. et al. *Extreme Programming Refactored: The Case Against XP,* APress L. P., 2003.

Talmy, L. 1996. The Windowing of Attention in Language. M. Shibatani, S.A. Thompson (ed.), *Grammatical constructions their form and meaning*. Oxford: Clarendon Press.

Toft, D. *E-journaler giver læger problemer,* Computerworld Online, 24. oktober 2003. Retrieved 17-6-2008: http://www.computerworld.dk/art/21231.

Tomayko, J. & Herbsleb, J. 2003. How Useful Is the Metaphor Component of Agile Methods? A Preliminary Study. CMU-CS-03-167. Carnegie-Mellon University.

Wake, W. 2000. *Extreme Programming Explored*. Boston: Addison-Wesley.

Way, E.C. 1991. *Knowledge Representation and Metaphor.* Kluwer.

Wenger, E & Lave, J. 1991. *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press.

West, D. 2002. Metaphor, Architecture and XP. In: In *Proceedings of the Third International Conference on Extreme Programming and Flexible Processes in Software Engineering (XP2002)* (Alghero, Italy, May 26--30 2002).

Øhrstrøm, P. 2003. Mellem Tanal og Gratex – et skoleeksempel inden for Humanistisk Datalogi. In. Schärfe, H. (ed.) *Impact*. Aalborg Universitetsforlag.

Øhrstrøm, P. et al. 2005.What has happened to Ontology. F. Dau, M.-L. Mugnier, G. Stumme (Eds.): *ICCS 2005,* LNAI 3596, pp. 425–438, 2005.Springer-Verlag Berlin Heidelberg.