Mind the Semantic Gap

David E. Millard
IAM Group
University of Southampton, UK
dem@ecs.soton.ac.uk

Nicholas M. Gibbins
IAM Group
University of Southampton, UK
nmg@ecs.soton.ac.uk

Danius T. Michaelides
IAM Group
University of Southampton, UK
dtm@ecs.soton.ac.uk

Mark J. Weal
IAM Group
University of Southampton, UK
mjw@ecs.soton.ac.uk

ABSTRACT

Hypertext can be seen as a logic representation, where semantics are encoded in both the textual nodes and the graph of links. Systems that have a very formal representation of these semantics are able to manipulate the hypertexts in a sophisticated way; for example by adapting them or sculpting them at run-time. However, hypertext systems which require the author to write in terms of structures with explicit semantics are difficult/costly to write in, and can be seen as too restrictive by certain authors because they do not allow the playful ambiguity often associated with literary hypertext.

In this paper we present a vector-based model of the formality of semantics in hypertext systems, where the vectors represent the translation of semantics from author to system and from system to reader. We categorise a variety of existing systems and draw out some general conclusions about the profiles they share. We believe that our model will help hypertext system designers analyse how their own systems formalise semantics, and will warn them when they need to mind the Semantic Gap between authors and readers.

Categories and Subject Descriptors:
H.5.4 [Hypertext/Hypermedia Theory]

General Terms: Theory

Keywords: Hypertext Formality, Hypertext Semantics

1. INTRODUCTION

From its earliest inception Hypertext has been considered a mechanism for aiding people in knowledge tasks and augmenting human thought [8, 12]. This was considered achievable because hypertexts act as a high-level formalism for modelling and communicating knowledge, in effect they are a logic representation.

Research has indicated that human authors struggle with highly formal systems. The low level of formalism in hypertexts can thus be seen as an advantage, allowing human authors to easily express their ideas and in more creative cases, such as literary hypertext, play with the ambiguity of meaning.

Despite this there is a growing trend in hypertext systems towards a 'translation to hypertext' approach. where knowledge is more formally encoded (for example, in an ontology) and then converted to hypertext at runtime. This reduces hypertext to an interface rather than a genuine way of structuring information and runs the risk of disenfranchising authors who cannot, or do not wish, to express themselves in a more formalised way.

In this paper we present a model that describes the formality of semantics in hypertext systems in such a way that they can be compared and common approaches identified.

We first explore the background of hypertext as a logic representation, and the previous work that has been done on the effect of formalisms on human authorship. We then set out our definition of semantics within a Hypertext, distinguishing between semantics conveyed in the content (the text) and that conveyed in the link structures themselves.

We characterise the communication of these semantics using a matrix of formal and informal author and readership, and examine the boundaries where hypertext changes in its intended use from one area of the matrix to another.

It is our belief that there is a tension within the matrix according to the extent of the grammar (or understanding) shared between an author and a reader. The further the system is from a shared grammar the greater the need for *mediation* (for example, the translation process described above).

Finally we present a vector model of semantics in hypertext systems based on reader/author formality and system mediation. We then use this model to describe a number of existing hypertext systems (including the Web, AHA!, StorySpace and the Semantic Web), and generalise some of the more common vectors in order to categorise system configurations that will find formality problematic.

Although we do not think that there is a magic solution to the problems of having different levels of semantic formality

HT'05, September 6–9, 2005, Salzburg, Austria.

in hypertext, we believe that our model will help system designers evaluate the way in which their own systems handle this formality, and warn them when they need to manage the Semantic Gap between the authors and readers that use them.

2. BACKGROUND

Different logic representations may be categorised according to their level of *formality*. In this paper we use the term formality to describe the rigidity of form and thus unambiguous precision of a particular logic representation, rather than how easy it is for a machine to extract useable knowledge from it (although in practise these are related).

Texts may be viewed as one of the less formal representations as they are flexible in form, forgiving of ambiguity and open to interpretation. Hypertexts offer the possibility of added formalism, for example by enforcing a restrictive set of link types, but are still less formal than strict concept graphs or other representations such as predicate logics.

Hypertext systems can therefore be seen as presenting an alternative to the type of full-blown knowledge representation used in natural language processing (such as frames, etc.) [22].

Early hypertext implementers exploited the representational power of hypertext structures to describe semantics. TextNet was an early hypertext system that treated texts as nodes in a semantic graph. Its aim was text organisation for the scientific community. [28]. Thoth-II was a system that also modelled a concept graph and optionally allowed text to be attached to the concepts, thus forming a hypertext. It is the text labels on links (simple typing) that means the hypertext can be viewed as a semantic net [10]. Other conceptualisations, such as Garg's hypertext abstractions, were based on set-theory [17].

Rich Hypertexts have been defined as hypertext networks made up of entirely typed nodes and links [25], and other researchers have also looked at typed anchors (taking into consideration the anchor's context of use) [23]. A rich hypertext not only helps in presenting a hypertext to a reader but can also be used to aid hypertext query, for example the use of Gram Path algebra to implement the semantic query of typed hypertexts in Multicard [2].

The rise of the web has increased the use of hypertext as a less formal logic representation, and encouraged its use as an extension of ordinary writing (perhaps because the web does not enforce node or link types, and offers no mechanisms for inference or rule-based navigation).

More recently the Semantic Web has emerged as a much more formal tool. The Semantic Web is designed explicitly for machine to machine communication and its logic representations, ontologies represented and described using languages such as RDF and OWL, are accordingly formal and restrictive.

Perhaps because of the two webs there is now a trend to create systems that follow a *translate to hypertext* approach to producing content and links. These systems use very formal logic representations to describe their knowledge and then translate these to less formal hypertext representations so that they can be easily understood by human users. This has the advantage that the formalised knowledge representations can be more easily managed and manipulated by the machine (for example, maintaining consistency, or adapting information according to some user model).

For example, Crampes and Ranwez translate relationships within an ontological knowledge base into web links [11]. Similar work has also been done within the Open Hypermedia community, such as using open hypermedia nodes that are queries into an ontological space that are resolved at runtime [30], or open hypermedia linkbases that are derived from an ontological knowledge base [9].

This is a paradigm that works well for the human as reader, but the original hypertext pioneers saw hypertext work as a human to human activity, which leads us to wonder about how such systems support the human as author?

Human authors do not always seek perfect clarity in their texts and often exploit the informality of written language; purposefully leaving room for readers to interpret what they have written. Systems that require their authors to express their ideas in such formal logics rob them of this ambiguity capability and actually reduce their ability to communicate as human beings, rather than augmenting it.

In 'Formality Considered Harmful' Shipman et al identified four key problems with human authors working with formalised information models [14] (examples are our own):

Cognitive Overhead Hypertext relationships have a larger granularity than those found in more formal knowledge representations such as an ontology. This means that users expressing knowledge in more formal ways are required to specify many small relationships which increases the effort necessary to conceptualise what they wish to express (for example, Falquet et al translate a chain of ontological triples into a single hypertext link [13]). Even once they have created it, many users still experience difficulties maintaining this kind of semantic network [29].

Premature Structure Hypertext ambiguity allows authors to evolve or grow structures (a feature that is leveraged for information triage in spatial hypertext systems [19]). Increased formalism may force them to specify their knowledge before it is fully formed.

Tacit Knowledge Domain experts are not always aware of their own knowledge, and may be unable to express it in a fully formalised manner. In the Knowledge Management community the first phase of the Knowledge Life Cycle [16] is Knowledge Acquisition, a structured process where knowledge experts observe and interview domain experts to extract such tacit knowledge.

Situational Structure There is also an argument that highly formal representations tie the knowledge too tightly to a particular task or situation. In these cases ambiguity could make it possible to use the information in a different context (because it could be interpreted differently).

There is therefore a tension in hypertext system design between highly formal knowledge representations that a machine might manipulate and less formal hypertext representations that human authors create and human readers wish to see. This situation is confused as hypertext structures themselves may be formally described using a machine friendly representation so that they can be communicated between software components. This machine-oriented representation formally describes the actual structures (for example, the relationships between anchors and links) rather than the intended meaning of the structures as a whole, and is more related to syntatics than semantics.

Trellis was a Petri-net based hypertext approach that not only included the hyperlinks and documents in its formalism, but also the windows and buttons of the hypertext system [27]. Trellis did not encode the semantics represented by the hypertext relations but the semantics of the hypertext mechanism itself.

In this paper we are not concerned with the formality of the mechanism description (the hypertext syntactics) but the formality of the semantics encoded in the hypertext structures and content.

Our goal is to present a model of hypertext systems that encapsulates the formality of these semantics and allows them to be analysed and compared.

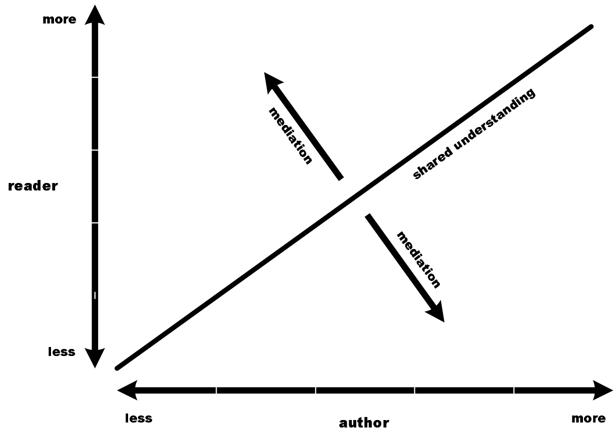

Figure 1: Mediation between author and reader formality.

3. EXPRESSING MEANING AND THE NEED FOR MEDIATION

If one ignores the semantics of the hypertext mechanism, then what is left is the meaning of hypertext structures and the meaning of the content that they interconnect. It is not our intention in this paper to explore either of these aspects too deeply, but we will give a few examples in order to demonstrate that the consideration of both is sensible, and to give the reader some concrete examples of the type of semantics that we are referring to.

Textual semantics is a complex subject, especially when the visual characteristics of written text is also taken into account. The meaning that a reader receives from a text is not necessarily the meaning that might be assumed from the literal semantics of the individual words used, or even the sentence constructions.

Writers play with word senses, for example using homonymy or synonymy, to influence the interpretations of their work.

They may also use tropes; phrases or words employed in a way that is incongruous with their original intended signification. For example using metaphor to emphasise the commonality between two seemingly unalike things ('the winter of our discontent...'), or using metonymy, perhaps to control emphasis ('The suits on Wall Street...').

Semantics is also implicitly contained in the presentation and arrangement of text. Typography can be used to convey meaning, such as using bold or italic fonts as a device to indicate emphasis, and the arrangement of textual elements can effect their reading. For example, a research paper such as this has a certain expected format (abstract, introduction, etc) which not only helps organise the information contained within, but can also effect the interpretation of the text (consider for example, the effect of labelling a section as 'Conclusions'). Writers can also play with layout to help express their ideas, such as Nelson's "Computer Lib/Dream Machines" [24] which is a physical embodiment of Nelson's notions of intertwingled text and, as two books in one, alludes to the joining of the two separate worlds described in the text.

Hypertext structures, which might be thought of as an advanced arrangement of text, can also have implicit semantics beyond any notion of links as simple conjunctions or link typing that is formally understood by the hypertext system.

To analyse the meaning in a hypertext it is necessary to have a notion of structures within the hypertext that are larger than simple links, or node/link pairs, but smaller that the extents of the entire hypertext graph. To help do this the notion of *Contours* was borrowed from the vocabulary of painting by Mark Bernstein, Michael Joyce and David Levine:

> "By contour, we mean a perceived pattern of meaning within the hypertext an observed structure that a reader can describe and appropriate for her own use." [6].

Contours are not just paths through a hypertext, although paths may follow contours. Joyce continues to take the more poetic view that contours are fluid shapes that coalesce in the reader's mind after they have been experienced [18]. Rosenberg describes Hypertext *Episodes*, as a part of a user's activity in hypertext that form a cohesive entity in the readers mind, he believes that contours form the geography of the hypertext episode [26].

Bernstein went on to take a more functional view and later identified a number of *patterns* that occurred regularly in hypertexts [3]. Patterns are recognisable hypertext structures such as forks or cycles, Bernstein believed that cycles in particular were crucial to contours, which form where cycles impinge on one another.

While these analyses give us a vocabulary in which to frame a discussion of hypertext semantics, it is clear that hypertext semantics are fluid and dynamic. Miles suggests that hypertext is fundamentally a temporal and a contextual experience, and that an essential part of that context is the development in the reader's mind of a narrative schema, which implies that a hypertext's principal meaningful structures are defined retrospectively [20].

It is thus clear that both text and link structures contain meaning, and that when read as a hypertext they come together as a semantic whole, although that meaning is dependant on each particular reader and the context of each reading.

For our inquiry into how hypertext systems formalise and transform semantics the interesting point seems to come when we consider the relation between the hypertext semantics intended by the author (created in the system) and those perceived by the reader (presented by the system), and how this translation is handled by the system.

We can characterise these relations in a simple two-dimensional matrix that maps the formality of authorship (the level of explicit semantics that the author specifies to the system during the creation process) and the formality of readership (the level of explicit semantics presented to the reader by the system during the reading process).

Figure 1 shows this matrix. The axes represent the extents of the formality of semantics. While we do not wish to place systems precisely into the resulting space we can say that generally formal systems, such as the Semantic Web, would appear in the top right part of the space and very informal systems, such as one that supports free linking, would appear in the bottom left.

These two points form the ends of a line that dissects the space. The line is where the level of formality of reading and of authoring is the same; authors and readers using systems on this line have a shared grammar, and a common understanding of how semantics should be expressed.

Systems that lie off of this line need to introduce a level of mediation so that their readers can understand the authored material. In effect they transform, or translate, the hypertext.

The most common systems off of the line seems to have more formal authorship than readership (they would be placed below the line). Many adaptive systems, such as AHA! [7], take this approach (the complex underlying semantics created by the author are rarely exposed to the reader). This allows the author to formally specify the adaptation in a conceptual fashion, but show the resulting hypertext structures to the reader in the usual way. It is the 'translation to hypertext' method described in Section 2.

Systems above the line are far less common, a kind of 'translation from hypertext' approach. These systems allow informal authorship, but present the resulting structures to the reader in a formal way. Shipman's Hyper-Object Substrate (HOS) is an example of this [15]. The HOS is a way of users expressing information informally and then incrementally refining it into a more formal structure. The aim is to address two of the problems identified with working with formalised information, premature structuring and excessive cognitive overhead.

In the next section we will simplify our representation of author/reader formality into a simple scale, and introduce a third point of representation that corresponds to the systems own model of the semantics (as separate to that which is exposed to either the author or the reader). In this way we can characterise the formality of any hypertext system as a pair of vectors along the scale.

4. A VECTOR-BASED MODEL OF FORMALITY IN HYPERTEXT SYSTEMS

It is possible to simplify our matrix into a single dimension of formality, and place both reader and author on this linear scale.

The scale is qualitative rather than quantitative, with the left end representing the least formal representations and the far right representing the most formal representations as shown in Figure 2.

As we move from left to right along the line, the explicitness of structure and meaning increases. Starting at the far left we have things such as plain documents. These documents by their very nature have structure, but it is implicit (as far as the system is concerned), as is the meaning of those structures.

Documents can be marked up with explicit structure, examples being HTML or LaTex. Here the structure is machine readable.

There may be additional meaning associated with these structures (beyond the semantics of the structure itself) but that is not made explicit to the system. For example, a chapter may be identified in LaTex with the title 'Prologue'. The system can understand the fact that this represents a chapter but the semantics of the chapter as a prologue is implicit in the title.

Moving further to the right, we can encapsulate explicit meaning within the structure alongside the document, for example by having strongly typed links in a rich hypertext.

Finally, as we reach the far right of our scale, the documents are replaced entirely by statements of explicit structure with explicit meaning: the vision of the Semantic Web.

So far we have described the scale by focusing on examples of formality levels along the scale. We can now place the two principle participants in the hypertext, the author and the reader, onto the scale and can begin to express the level of formality presented to each of them.

Systems which have a shared grammar (exist on the line in Figure 1) have a reader and author on the same point of the scale. Systems which require mediation have them apart. The author to the right of the reader in cases of 'translation to hypertext', and to the left in cases of 'translation from hypertext'.

This is still not a complete view of the way that semantics are transformed within a system. The final element we wish to place along the scale is that of the system itself. Often the structures maintained by the system are different from those created by the author during the authoring process and different to those presented to the reader during the reading process, this is because the system can have its own independent semantic requirements according to how it intends to manipulate the structures. Figure 3 shows our final scale, with all three points shown.

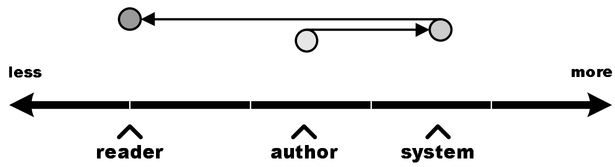

Figure 3: Final Vector Scale

A number of things can quickly be noted from this form of representation. The position of the points along the line gives an indication of how formal the overall system is. With all the points at the far left, the system does not deal with explicit structure or explicit meaning. With all the points at the far right of the scale, the system deals solely in these properties with the authoring process requiring their explicit creation and the reading process involving the manipulation of explicit structures and the communication of explicit meanings.

We thus model the formality of semantic representation within a hypertext system as the movement from author, via authoring tool or interface, to system, and then from system, via reading interface, to the reader. The magnitude of the vectors provides an indication of mediation. This

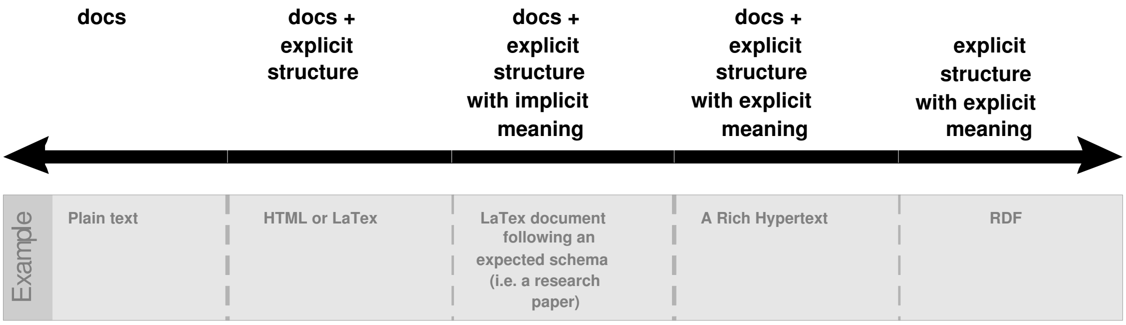

Figure 2: Single Scale of Formality

could be the inference of explicit structures and meaning from less formal authoring or could be the publishing of explicit structures in a less formal manner for the reader.

The model does involve elements of simplification. In practice, systems may well have some flexibility in the formality of semantics represented to authors and readers and thus these users would sometimes be best represented as a range on the scale, rather than a point. A complex system may also have multiple internal semantic representations (for example, as structures are passed from one process to another). This would result in more than one system point and thus more than two representative vectors for the model as a whole. Finally it is not impossible for some systems to have multiple types of reader or author with different levels of formality, however in practise this seems rare.

5. EXEMPLAR SYSTEMS

In this section we briefly categorise a number of exemplar hypertext systems using our vector model; we have chosen to use a single representational point for authors and readers for reasons of clarity.

We have chosen these systems as we feel that they will largely be familiar to the hypertext community and also as they display a wide variety both in terms of hypertext domain and also complexity of functionality.

5.1 Web

Figure 4: The Web

The Web obviously covers a wide range of possible applications but for the purposes of this example we are looking at authors writing basic HTML which is then read by readers in a standard web browser.

Figure 4 shows how a basic Web system might be categorised. The authoring involves writing a document with some explicit structure but with no statements of meaning. The system holds these documents and structures without any modification. When being presented to the reader, some of these explicitly authored structures are translated into the document itself losing even the semantics of the structuring (for example, headings which are only identifiable by the reader based on inferences about font style and size). However, the reader is still presented with explicit structures to manipulate in the form of hypertext links so the Author, System and Reader can all be placed at approximately the same place on our formalism scale.

5.2 Semantic Web

Figure 5: The Semantic Web

Figure 5 shows our placements on the scale for a typical semantic web application. Strictly speaking the roles of author and reader are being stretched here since, as we have already stated, the semantic web is primarily a mechanism for machine to machine communication. The 'authoring' process often involves the automatic translation of existing data into RDF triples which are stored. 'Reading' involves the processing of these triples. We have placed all three points, Reader, System, Author at the far right end of our scale as in its purest form the semantic web has no documents, only simple structure with explicitly stated meaning.

In practice, many semantic web applications will involve the inference of structure and meaning from existing information, or the translation of structure and meaning into more human parsable form. However, we believe that such approaches merit independent characterisations of their own.

5.3 Storyspace

Storyspace [5] has become the hypertext tool of choice for creators of published hypertext fiction. One of its strengths is that the interface for authoring is similar to the interface for reading. Graphical tools for linking and organising material reconstitute themselves as maps for readers. The structure is explicitly authored in the form of links although these do connect media nodes which will of course contain structure of their own.

Figure 6: Storyspace

In Figure 6 the author and reader are placed at the same point along the scale, with both working with explicit structures with no stated meaning (formal or otherwise). The authors do have the opportunity to use features such as guard fields to apply additional structuring to the links but again, the meaning of these guard fields is not explicitly encapsulated during the authoring process. The system, as in the Web example, exists at the same point as author and reader.

5.4 AHA!

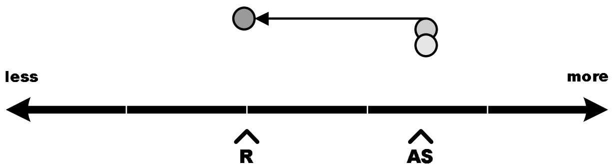

Figure 7: AHA!

The AHA! hypermedia system [7] has an authoring process much more formal than the reading process (Figure 7). When authoring AHA! applications documents are created along with explicit structures with explicit meanings. Information is structured around hierarchies of concepts which are then used by the system to tailor the reading experience. However, these structures are not all translated to the reader during their reading process (the concepts driving the adaptation remain hidden), and those that are often loose their explicit semantics and are represented through adaptive techniques such as stretch text or link annotation.

5.5 Card Shark

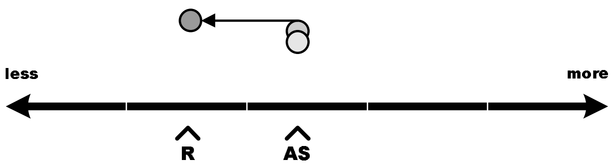

Figure 8: Card Shark

Card Shark is a hypertext system created by Bernstein as an example of what he calls 'strange hypertext' (Hypertext paradigms that stretch our notions of what it means to be a hypertext) [4]. In Card Shark a player is dealt a number of cards containing rules, conditions and lexia. When they play a card (read it) they invoke the rules on the card which then changes the state of the system, their hand is then replenished. The state is then compared to the conditions on all the cards in their hand and only those cards that are compatible with the state are playable. This is termed Sculptural Hypertext (as compared to normal Calligraphic Hypertext) as all the lexia (the cards) are initially linked, and certain links are effectively removed at every stage of reading (according to the current cards in the players hand and their conditions compared to the system state).

Because Card Shark was an experimental exemplar system it has no real authoring environment, so the author and system formality are the same. However, it is interesting to compare it to AHA! as both systems are doing some system reasoning at a higher level of formality than the reader is reading. In the case of AHA! the author specifies the adaptation of the system in a *programmatic* way, thus the behaviour, but not the meaning is specified. In comparison, authoring in Card Shark is *declarative* as meaning is attached to cards in the form of simple rules and conditions, however this meaning is implicit because the declarations are not themselves formalised (there is no strict vocabulary or ontology).

5.6 ArtEquAKT

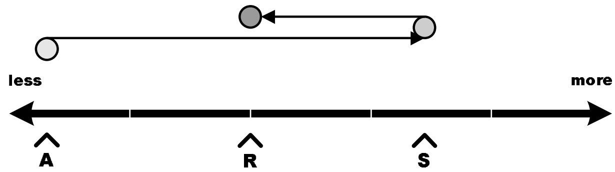

Figure 9: The ArtEquAKT system

The ArtEquAKT system [1] generates Biographies of artists from information gathered from the Web. Not strictly a hypermedia application in the traditional sense it does use both hypermedia and semantic web technologies as part of a larger composite architecture.

Figure 9 shows quite a distance in placement between the three noted points on the scale. The 'authoring' process involves extracting information from documents on the web (the text rather than the networks of links). The distance between Author representations (basic web pages and text) and System representation (explicitly structured knowledge with linked textual fragments) represents the process of knowledge extraction that is taking place. Implicit structure and meaning in the web document's text is being automatically extracted and recorded into explicit knowledge as an ontologically structured knowledge base. When this is presented back to the reader, the knowledge is combined and published as a hypermedia document with a range of adaptive features. This process provides document and explicit structure to the reader and some explicit meaning becomes implicit in the content of the document produced (via stretching and dimming text).

5.7 VIKI

VIKI brings a different aspect to our analysis as it is a spatial hypertext system [19]. Users of the system group lexia spatially using a graphical interface. VIKI analyses these spatial arrangements to create more explicit structures that could then be presented back to other users of the system. We have chosen to categorise this system in Figure 10 by placing the author towards the left of the scale as the grouping of nodes is communicating implicit structure which is concretised by the system in its storing of the

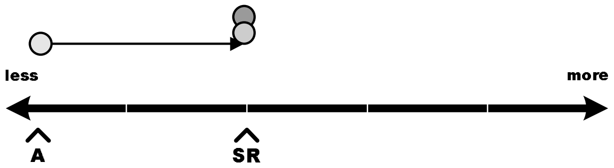

Figure 10: VIKI

hypermedia structures and may then be named, providing implicit meaning. These more explicit structures are then presented back to the readers of the systems.

6. GENERAL ISSUES

Having looked at a variety of different systems from different domains of hypermedia we can begin to see some general issues emerging.

Firstly, the processes of *knowledge capture* and *natural language processing* can be identified along the scale. Knowledge capture involves the creation of explicit structure and semantics from less formal documents. In the ArtEquakt example above this is the extraction of facts from simple web documents. This process is not 100% accurate however, with ambiguity of meaning in the text and limitations of extraction techniques. The further apart the authoring tools are from the formal representations held by the system, the more chance for the introduction of error in this translation process.

Similarly, *natural language generation* translates often very formally specified structures with explicit meanings into more freeform structures, usually text. Here too, the translation process can introduce errors and the users of the end product do not have the richness of structure held by the system to manipulate.

One of the reasons that the ArtEquAKT system uses a more formal internal representation than it shows to readers, is that the formal representation allows it to reason about these errors (in this case by consolidating the knowledge base).

Another issue to draw from the examples above is that typically the further to the right the authoring process moves, the more complicated, and often expensive, the authoring process becomes. This is formality considered harmful [14]. Authoring in systems with simple formality of structure (Web, Storyspace, VIKI) is relatively straightforward, and in the case of VIKI some of the strain is being taken up by the system as it translates the implicitly defined structures of the author into more explicit, machine manipulable, structures to be held internally. Semantic Web technologies often require very detailed authoring, the creation of ontologies, and complex authoring tools if authors are to produce semantic data manually. Of course this does reflect the intention for it to be a machine to machine communication.

The difference between the programmatic AHA! and the declarative Card Shark is also an indicator of how formality of semantics can result in a loss of control for the author. In previous work we have described this conflicting authoring strategy as a tension between freeform and designed hypertexts [21]. Systems that follow a freeform method (such as sculptural hypertexts) are likely to have a more formal system representation of semantics than those that are designed, as they need to reason about the lexia at run-time.

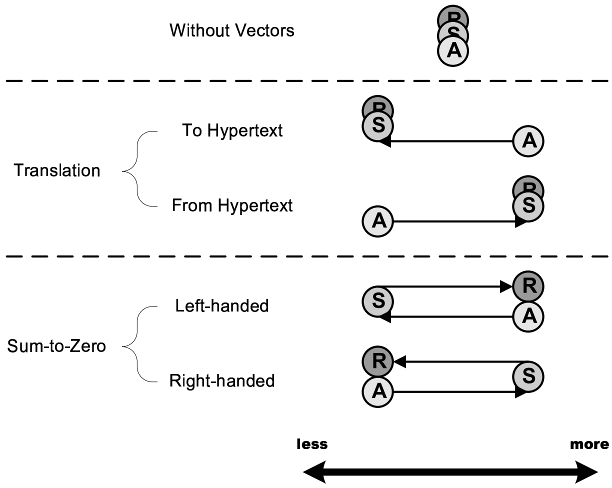

Figure 11: Common Profiles

We can summarise by saying that systems that are popular with writers (e.g. StorySpace and the Web) tend to have both informal reader and authorship, as writers may wish to play with semantic ambiguity, and believe that readers should be free to interpret their work in the reader's own way.

Systems that wish to manipulate or adapt the structures in some way have an internal representation that is rather more formal (for example AHA!). But may run into problems engaging with writers who do not wish to formalise their creations. If the system has readership that is less formal that the system representation then authors may also be worried that the meaning of their work is in some way changed when it is translated to be presented to the reader.

Some systems may try to solve the problem of writers who do not wish to formalise their work by having informal authorship and then translating to a more formal system representation (such as VIKI or ArtEquAKT), but these still have the problem that the process of mediation or translation may alter the original intent of the author. Although we should acknowledge that neither system poses as a writers tool; VIKI is intended as an information triage tool, and ArtEquAKT a knowledge extraction tool.

7. COMMON PROFILES

Section 5 looked at how several hypertext systems map onto the vector model. In this section we shall attempt to draw out some of the common profiles and discuss their characteristics. Figure 11 shows the profiles we have detailed.

7.1 Without Vectors

Systems with no vectors (author, system and reader have the same formality of semantics) demonstrate a genuine shared grammar between their participants, i.e. authors write at the same level of formality as the system processes and the reader reads. It is possible that these formalisms may be comparably formal, but different formalisms, this was not true for any of our exemplars, but would have an effect similar to a sum-to-zero vector (see Section 7.3 below).

7.2 Translation Vectors

Translation Vectors occur in systems with an element of mediation, that translate from one level of semantic formality to another.

Left-facing System-Reader vectors indicate a system that takes a 'translation to hypertext' approach (moving from a more formal system representation to a less formal reading one). This process is analogous to natural language generation. It is an approach favoured by adaptive systems such as AHA! and is also important for sculptural hypertext systems similar to Card Shark.

Right-facing Author-System vectors indicate a system that takes a 'translation from hypertext' approach (moving from a less formal human authoring model to a more formal system one). This process is analogous to knowledge capture (for example, natural language processing). It is an approach favoured by evolving-knowledge tools such as VIKI.

7.3 Sum-to-Zero Vectors

Systems with vectors that sum-to-zero (author and reader have the same formality, but the system is different) appear to participants to have a shared grammar but actually semantics may be altered through the system. This may be important for authors whose work is sensitive to the grammar (for example, hypertext authors who are constructing their own grammar within the scope of a particular piece of work). Although the sum is not quite zero, ArtEquAKT takes a right-handed version of this approach (the system is more formal that author or reader). It seems suited to systems that wish to manipulate the knowledge at a high level that is inappropriate to expose to human participants.

It is hard to imagine a left-handed sum-to-zero system, this would demand the formal creation of semantics from its authors, throw this away in the system representation, and then be forced to rediscover them algorithmically so that the structures could be presented back to a reader. However, the individual vectors do make sense. A left-facing Author-System vector (formal authorship to less formal system model) might be useful if the purpose was to help the author structure their work (for example, writing an essay with the aid of Toulmin structures). A right-facing System-Reader vector (informal system model to a formal reader presentation) might also be useful if the readers task was correspondingly formal (such as using a Toulmin analysis tool to parse a written essay). Although we did not look at any existing systems that fit these profiles it is only their predesigned coupling that seems unlikely.

8. CONCLUSIONS

In this paper we have looked at the ways in which hypertext can be considered as a logic representation and explored how semantics can be encoded in its structure. We have explained that this can be done textually, in the content and presentation of the hypertext nodes, but also how it can be conveyed in hypertext links, in forms such as contours, episodes or patterns.

We have presented a vector-based model of how semantics are formalised in hypertext systems. There are (at least) two vectors: the first represents the difference in the formality of the representation of semantics between authors and system, the second represents the difference between system and readers. This model is capable of incorporating systems as diverse as the web and the semantic web, and covers work from ontological knowledge bases to literary hypertexts. It has allowed us to compare various system's approaches and pull out general issues based on several common profiles that existing systems share.

It is the Left-facing System-Reader vector, the 'translation to hypertext' paradigm, that seems the most problematic of our profiles. At the moment, perhaps because of the Semantic Web, it is attracting a lot of attention from the hypertext systems community and yet seems to disenfranchise human authors who wish to exploit the semantic ambiguity of hypertext structure. It is not just the harm of formality itself that causes the problems, but the fact that some authors may also want to play with the grammar they share with their readers, which is difficult in mediated systems.

The 'translation from hypertext' paradigm may make formality seem less harmful, but it is still mediation and thus continues to cause problems for authors who wish to write ambiguously.

Formality is also not always an unnecessary evil, even from the writers point of view. Sculptural hypertext is a declarative model that depends on authors stating some semantics (although this may be implicit semantics) to drive the sculptural mechanism. Without the declarations there is no sculpture.

It has not been our intention in this paper to generate design patterns for hypertext formality, nor even to suggest that the profiles we have presented are more appropriate for one type of application than another in any absolute way. Instead we have presented a framework in which the formality of semantics in a system can be discussed and systems compared.

Unfortunately any hypertext system that wishes to manipulate its hypertext structures in a sophisticated manner needs to have some kind of formalised view of hypertext semantics. But if authors wish to access this power, for example to create large adaptive hypertexts or explore sculptural hypertext writing, it seems that they will need to relinquish some of the control of their own creations. At the same time creators of formalised systems should not assume that the original authored structures can be understood fully in a presubscribed manner, and should be sensitive to the ways in which their system's formalisms could effect hypertext writers.

We do not believe that there is any magic solution to this problem. In any system where there is an intermediatory between the author and their audience, we should expect interpretation. This occurs in music and theatre and is understood and appreciated by participants.

Authors, readers and system designers in the hypertext domain also need to appreciate this interpretation. They will always want to work at different levels of abstraction, and thus different degrees of semantic formality. Sometimes, depending on the profile of their system interaction, this will be difficult, but hopefully the model of semantic formality we have presented in this paper will help them to mind the Semantic Gap.

9. ACKNOWLEDGEMENTS

This research is funded by EPSRC IRC project "EQUATOR" GR/N15986/01.

10. REFERENCES

[1] H. Alani, S. Kim, D. E. Millard, M. J. Weal, W. Hall, P. H. Lewis, and N. R. Shadbolt. Automatic

ontology-based knowledge extraction and tailored biography generation from the web. In *IEEE Intelligent Systems*, pages 14–21, 2003.

[2] B. Amann, M. Scholl, and A. Rizk. Querying typed hypertexts in multicard/o2. In *ECHT '94: Proceedings of the 1994 ACM European conference on Hypermedia technology*, pages 198–205. ACM Press, 1994.

[3] M. Bernstein. Patterns of hypertext. In *HYPERTEXT '98: Proceedings of the ninth ACM conference on Hypertext and hypermedia : links, objects, time and space—structure in hypermedia systems*, pages 21–29. ACM Press, 1998.

[4] M. Bernstein. Card shark and thespis: exotic tools for hypertext narrative. In *HYPERTEXT '01: Proceedings of the twelfth ACM conference on Hypertext and Hypermedia*, pages 41–50. ACM Press, 2001.

[5] M. Bernstein. Storyspace 1. In *HYPERTEXT '02: Proceedings of the thirteenth ACM conference on Hypertext and hypermedia*, pages 172–181. ACM Press, 2002.

[6] M. Bernstein, M. Joyce, and D. Levine. Contours of constructive hypertexts. In *ECHT '92: Proceedings of the ACM conference on Hypertext*, pages 161–170. ACM Press, 1992.

[7] P. D. Bra, A. Aerts, B. Berden, B. de Lange, B. Rousseau, T. Santic, D. Smits, and N. Stash. Aha! the adaptive hypermedia architecture. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 81–84. ACM Press, 2003.

[8] V. Bush. As we may think. *Atlantic Monthly*, 176(1):101–108, 1945.

[9] L. Carr, W. Hall, S. Bechhofer, and C. Goble. Conceptual linking: ontology-based open hypermedia. In *WWW '01: Proceedings of the tenth international conference on World Wide Web*, pages 334–342. ACM Press, 2001.

[10] G. H. Collier. Thoth-ii: hypertext with explicit semantics. In *HYPERTEXT '87: Proceeding of the ACM conference on Hypertext*, pages 269–289. ACM Press, 1987.

[11] M. Crampes and S. Ranwez. Ontology-supported and ontology-driven conceptual navigation on the world wide web. In *HYPERTEXT '00: Proceedings of the eleventh ACM on Hypertext and hypermedia*, pages 191–199. ACM Press, 2000.

[12] D. C. Engelbart. A conceptual framework for the augmentation of man's intellect. *Vistas in Information Handling*, 1:1–29, 1963.

[13] G. Falquet, L. Nerima, and J.-C. Ziswiler. Towards digital libraries of virtual hyperbooks. In *HYPERTEXT '04: Proceedings of the fifteenth ACM conference on Hypertext & hypermedia*, pages 24–25. ACM Press, 2004.

[14] I. Frank M. Shipman and C. C. Marshall. Formality considered harmful: Experiences, emerging themes, and directions on the use of formal representations in interactive systems. *Comput. Supported Coop. Work*, 8(4):333–352, 1999.

[15] I. Frank M. Shipman and R. J. McCall. Incremental formalization with the hyper-object substrate. *ACM Trans. Inf. Syst.*, 17(2):199–227, 1999.

[16] B. Gallupe. Knowledge management systems: surveying the landscape. *Int J Management Reviews*, 3(1):61–61, 2001.

[17] P. K. Garg. Abstraction mechanisms in hypertext. In *HYPERTEXT '87: Proceeding of the ACM conference on Hypertext*, pages 375–395. ACM Press, 1987.

[18] M. Joyce. *Of Two Minds*. University of Michigan Press, 1994.

[19] C. C. Marshall, I. Frank M. Shipman, and J. H. Coombs. Viki: spatial hypertext supporting emergent structure. In *ECHT '94: Proceedings of the 1994 ACM European conference on Hypermedia technology*, pages 13–23. ACM Press, 1994.

[20] A. Miles. Hypertext structure as the event of connection. In *HYPERTEXT '01: Proceedings of the twelfth ACM conference on Hypertext and Hypermedia*, pages 61–68. ACM Press, 2001.

[21] D. Millard, H. Davis, M. Weal, K. Aben, and P. D. Bra. Aha! meets auld linky: integrating designed and free-form hypertext systems. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 161–169. ACM Press, 2003.

[22] J. Nanard and M. Nanard. Using structured types to incorporate knowledge in hypertext. In *HYPERTEXT '91: Proceedings of the third annual ACM conference on Hypertext*, pages 329–343. ACM Press, 1991.

[23] J. Nanard and M. Nanard. Should anchors be typed too?: an experiment with macweb. In *HYPERTEXT '93: Proceedings of the fifth ACM conference on Hypertext*, pages 51–62. ACM Press, 1993.

[24] T. Nelson. *Computer lib/Dream Machines*. self published (reprinted by Tempus Books in an updated form in 1987), 1974.

[25] K. Østerbye and K. Nørmark. An interaction engine for rich hypertexts. In *ECHT '94: Proceedings of the 1994 ACM European conference on Hypermedia technology*, pages 167–176. ACM Press, 1994.

[26] J. Rosenberg. The structure of hypertext activity. In *HYPERTEXT '96: Proceedings of the the seventh ACM conference on Hypertext*, pages 22–30. ACM Press, 1996.

[27] P. D. Stotts and R. Furuta. Adding browsing semantics to the hypertext model. In *DOCPROCS '88: Proceedings of the ACM conference on Document processing systems*, pages 43–50. ACM Press, 1988.

[28] R. H. Trigg and M. Weiser. Textnet: a network-based approach to text handling. *ACM Trans. Inf. Syst.*, 4(1):1–23, 1986.

[29] W. Wang and R. Rada. Structured hypertext with domain semantics. *ACM Trans. Inf. Syst.*, 16(4):372–412, 1998.

[30] M. J. Weal, G. V. Hughes, D. E. Millard, and L. Moreau. Open hypermedia as a navigational interface to ontological information spaces. In *HYPERTEXT '01: Proceedings of the twelfth ACM conference on Hypertext and Hypermedia*, pages 227–236. ACM Press, 2001.