Knowledge Representation Graphs

From concept structuring to Web resource management

Limoges, March 12 2012

Tanguy Wettengel

PhD Computational Linguistics - Higher Doctorate in Language Sciences
University of Limoges (France) - Senior Corporate Consultant

tanguyjardin@icloud.com
*https://fr.**linkedin**.com/in/**tanguy**-**wettengel**-66624578*

Table of contents

1. Introduction 3

2. Knowledge representation diagrams 3

2.1. General purpose knowledge graphs 4

2.2. Addressable resource-connected knowledge graphs 9

RDF *9*

Topic Maps *10*

2.3. A brief comparison between Topic Maps and RDF graphs 14

3. Collaborative knowledge architectures 15

Wikipedia *16*

Fuzzzy *16*

References 17

1. Introduction

Organising the content of web resources to fit the meaning (and not the wording) of queries is the on the agenda of the future Semantic Web. To reach this goal, content-oriented metadata engineering via ontologies is becoming a standard, and benefits from formal language support (RDF, OWL, XTM). Besides text-centred interfaces such as Protégé, graphical ones (Onotoa, for example) equally allow to produce structured metadata by manipulation of boxes that contain text fields. As metadata are engineered to make content retrieval efficient by invoking meaning instead of just matching the syntax of expressions, knowledge structures encoding meaning become the cornerstone of resource filtering. The organic nature of such structures is enforced by linking together its primary entities, generally thought of as "concepts".

Within computing, considering knowledge as built out of concepts that are connected to each other in different ways, formally binds the structure of metadata to the Entity / Relationship paradigm. We consider the Entity / Relationship model as a paradigm because it acts as a framework for a variety of notations in information engineering, linguistics and computer science. In his well celebrated article [15], Chen explains that "The entity-relationship model adopts the [...] view that the real world consists of entities and relationships.". The data are seen from different perspectives (4 levels) in Chen's model. Level 1 appears as the most relevant to our purpose, because it contains entities and meaningful groupings of them into sets, as well as relations between them. Every set has a predicate that is true for entities belonging to it. Level 1 equally contains n-ary relations between entities and meaningful groupings of them into sets. Entities and relationships are described by attribute-value pairs, also grouped into sets.

Although aimed at database modelling, Chen's systematic analysis covers crucial questions related to what a concept and a relation might be in structures picturing knowledge. However, this relatively large framework has allowed for different varieties of data and knowledge-structure models to develop. As graphical renderings provide a picture allowing easy access to them for non-specialists, and as wikis are commonly aimed at users that are not experts in formal languages, we will here focus on knowledge representations that can easily translate into diagrams. Moreover, echoing the fact that Web culture is shifting from global to community-oriented, we consider situations where the organisation of knowledge mirrored by the metadata can be realistically managed by user groups that share a perspective on the subjects documented by the resources they handle.

The aim of this paper is to discuss different types of knowledge representations using graphical notation to express meaning, and further to suggest how to implement one of them for lightweight metadata and resource management by a user community. Section 2 deals with different approaches to knowledge diagramming, Section 3 with collaborative knowledge architectures, Section 4 with an implementation of a knowledge graph for structuring metadata and binding resources to them via a graphical interface.

2. Knowledge representation diagrams

Among the existing artefacts to express conceptual relatedness (which *de facto* include natural languages and formal ones, such as logic notations), graphical metaphors have earned much attention during the last 50 years (Semantic Networks, KL-ONE, Conceptual Graphs, Knowledge Maps, UML, Concept Maps, Mind Maps, etc. have been pictured in diagrammatic form). The variants interpreting the organisation of concepts as a meaningful distribution of linked points in some space (generally termed "maps") not only gain increasing popularity as being easy-readable but, in some cases, also claim to "naturally" reflect the psychological structure of knowledge. Examples of this claim have caused translating relatedness among concepts into a measurable distance between nodes (number of nodes to cross in order to reach a point from another one), or rendering of the generic - specific hierarchy by a top-down layout of concept nodes, or picturing associations of thought or sequences in discourse by branching links in a diagram.

As numerous tools provide a means to put knowledge diagramming swiftly at work, questions about the foundations of such claims and, particularly, what a node or a link really stand for, are frequently left aside. As to that, quoting Novak [1] a recent article of D. Ifenthaler [2] points out that "the development of new tools, instruments and methodologies to capture key latent variables associated with human learning and cognition requires a solid theoretical foundation" whereas "many of them are developed with little or no theoretical justification".

To gain some insight into these models, we first review different knowledge graphs by considering their the capacity in both the fields of concept structuring and spatial rendering of the resulting models. However, while describing them, we will not take the conventional distinction between "hierarchies" and "networks" - single or multiple parent allowance for a node - into account.

2.1. General purpose knowledge graphs

The attempts to set up diagrams representing concepts by means of nodes and links go back to the 3rd century, as witnessed by the tree figuring the enhancements of Aristotles's categories scheme, drawn by (or after) the phoenician philosopher Malc, known as "Porphyry" (see Figure 1 for a fragment of it). Sowa [3] points out that several foundational notions of contemporary conceptual diagramming are already in this type of representation (inheritance of properties from super-types to subtypes, "universal" / "particular" distinction).

Towards the end of the XVI century, the inventor of the term "Ontology" as separate from the philosophical theory of being, the German philosopher Jacob Lorhard, used left-to-right bracket diagramming to represent conceptual structure, mainly splitting types into exclusive subtypes ("either / or" subclasses) [4].

Discussing accessibility of diagrammatic representations, Corrin Gurr [5] underlines that "… the logical and spatio-visual properties of structures inherent to the diagram are chosen so as to have some very direct correspondence with the structures that they represent in the semantic domain". Larkin and Simon [6])share this perspective when comparing diagrams with natural language: "… the diagrammatic representation preserves explicitly the information about the topological and geometric relations among the components of the problem, while the sentential representation does not.".

When collectively managing metadata, successful knowledge sharing strongly depends on the choice of notations, i.e., on the the capability of representational means to allow an intuitive understanding of structure. We consider graphical notations as "expressive" when inferences are afforded [7] and controlled by graphical conventions (defining types of links and nodes) and when the labels (symbols, signs, phrases) conform to explicit semantic and syntactic well-formedness rules.

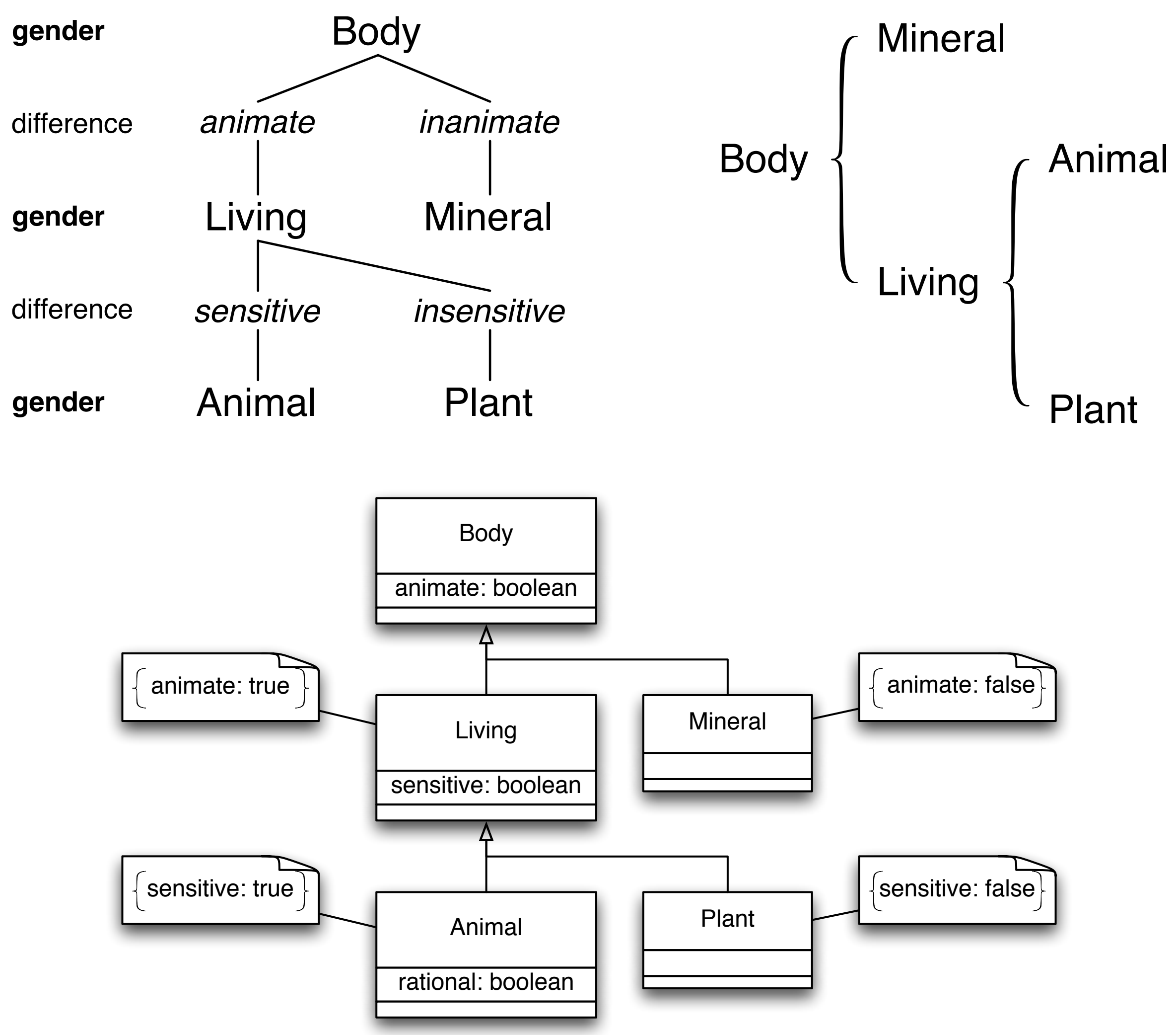


***Figure 1**: A fragment of Porphyry's tree (left), a simplified Lorhard diagram and a UML equivalent class diagram*

Discussing accessibility of diagrammatic representations, Corrin Gurr [5] underlines that "… the logical and spatio-visual properties of structures inherent to the diagram are chosen so as to have some very direct correspondence with the structures that they represent in the semantic domain". Larkin and Simon [6])share this perspective when comparing diagrams with natural language: "… the diagrammatic representation preserves explicitly the information about the topological and geometric relations among the components of the problem, while the sentential representation does not.".

When collectively managing metadata, successful knowledge sharing strongly depends on the choice of notations, i.e., on the the capability of representational means to allow an intuitive understanding of structure. We consider graphical notations as "expressive" when inferences are afforded [7] and controlled by graphical conventions (defining types of links and nodes) and when the labels (symbols, signs, phrases) conform to explicit semantic and syntactic well-formedness rules.

As metadata are aimed at resource classification purposes, we here restrict our review of knowledge graphs to what Sowa [3] terms "Definitional Networks" (as opposed to "assertional", "implicational" and "executable" ones). Regarding Selz's schemas, listed among Sowa's executable networks, we still understand them as (frame-type) definitional networks, even though their void slots imply mechanisms aimed to change their content.

Figure 2 shows one such schema as well as a fragment of a KL-ONE [8] representation, featuring properties enhanced by cardinality values to shape the definition of concepts. As compared with tree-flavoured layouts, the spatial arrangement of the KL-ONE diagram seems to disallow elementary reading strategies, mainly because horizontal and vertical sequencing do not appear as significant.

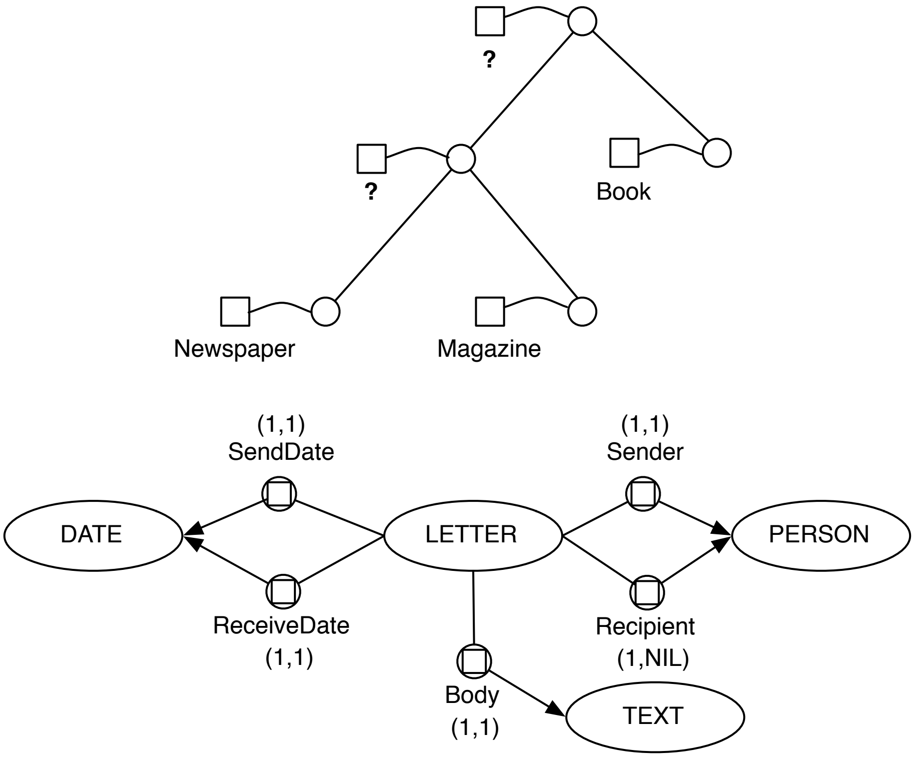


***Figure 2** : A diagram used by Otto Selz in his psychological experiments and a fragment of a KL-ONE definition*

Whereas capturing hierarchical dependencies through layout can visually imply inheritance of features, the semantic difference between links tying properties to concepts in representations such as Collin and Quillian's model of semantic memory [9] ("can bite" and "is dangerous" for the concept "Shark") does not translate into the diagramming language. An even less semantically loaded version of links is implemented by spiderweb (non-hierarchical) diagrams such as the Collins & Loftus network [10], which aims at translating relatedness of concepts by the distance between freely connected nodes.

Charging links with transparent semantics has been achieved both in a formal and an informal way (KL-ONE and Concept Maps illustrate opposite approaches to this question). In Concept Maps [11], links are non-directed (thus forcing the reader to discover the sense of the relation they establish). Regarding labelling, neither nodes nor links benefit from conceptual regularity, as may be noticed when comparing in Fig. 4 "Long-term Memory", a noun standing for an entity, with "Prompting by a Teacher", a nominalised verb phrase representing a situation.

Following a path in Concept Maps results in building full sentences and hence does not invoke the inferential capacities of the reader to generate propositional meaning himself. Yet, allowing notational economy, this is one of the key advantages of graphical constructs over verbose encodings of knowledge. As Larkin and Simon [6] put it, "Diagrams can group together all information that is used together, thus avoiding large amounts of search for the elements needed to make a problem-solving inference. Diagrams typically use location to group information about a single element, avoiding the need to match symbolic labels. Diagrams automatically support a large number of perceptual inferences, which are extremely easy for humans."

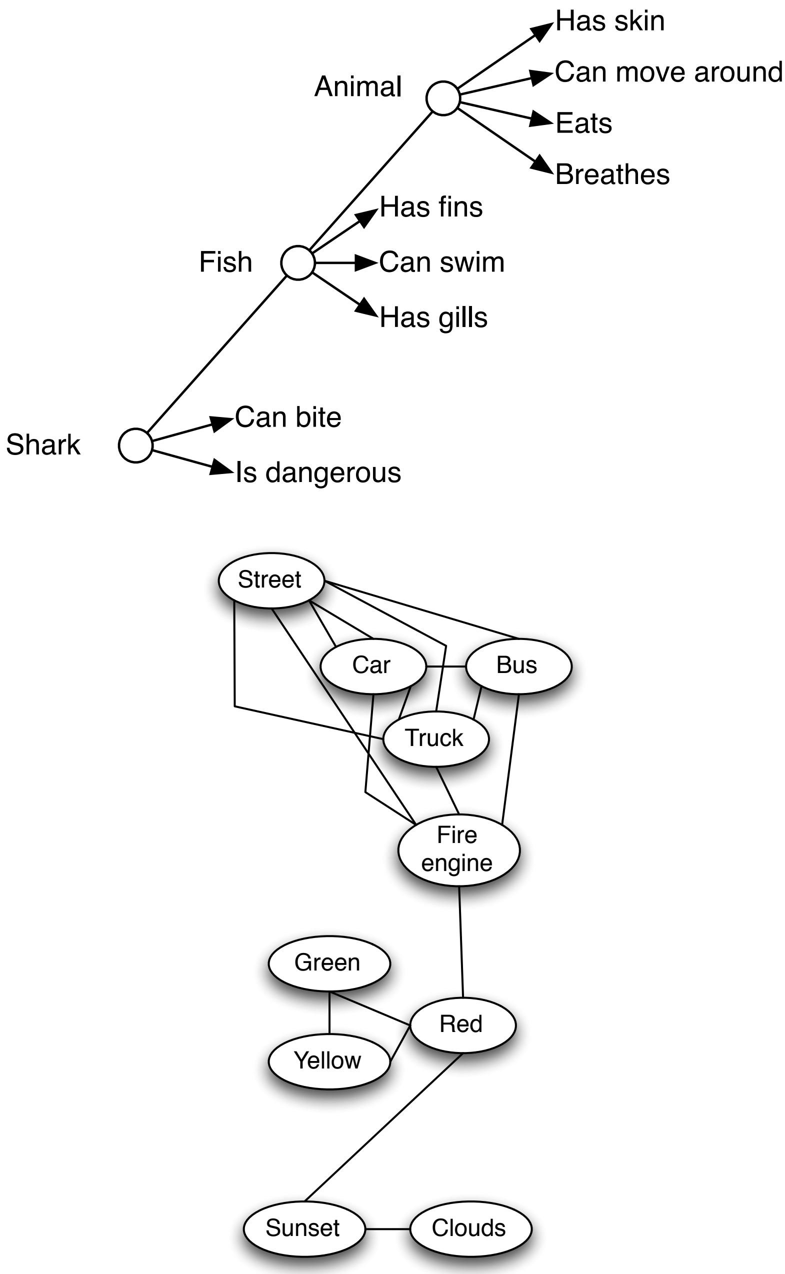


***Figure 3**: An example of Collin and Quillian's model, of Collins and Loftus network*

An attempt to introduce regularity and abstraction into labelling has been undertaken in McCagg & Dansereau's contribution to Knowledge Maps [12], where links are categorised into Dynamic (with subclasses such as "influences", "next" and "leads to"), Static (with subclasses such as "type", "part" and "characteristics") and Elaboration (with subclasses such as "example" and "analogy"). Figure 4 elaborates the content expressed by the example Concept Map as a Knowledge Map. Links are directed, labels P, T and C (Static links) stand for "part", "type" and "characteristics"; label L, for "leads to" (Dynamic link).

A different typology of links appears in a diagramming system aimed at educational purposes [13] and claiming to express "[…] what the brain does". Fig. 5 exhibits some type of concept-relating structures used in this rather informal framework, where instead of labelling links, concept organisation is tied to the overall morphology of graphs.

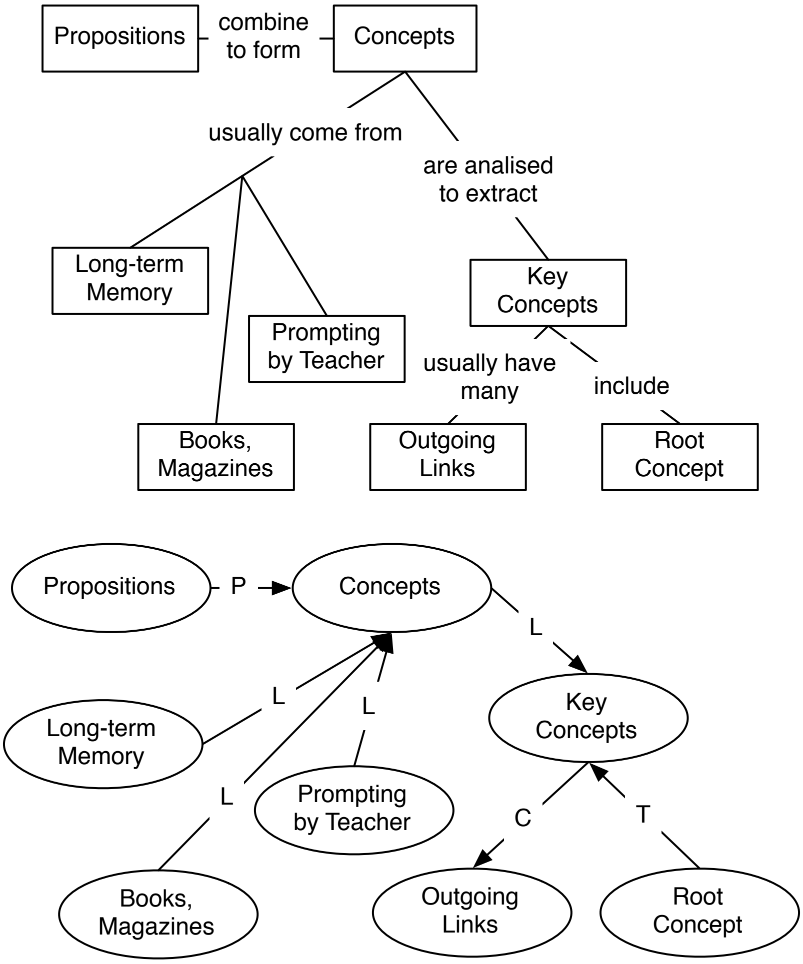


***Figure 4**: A concept map and the equivalent knowledge map*

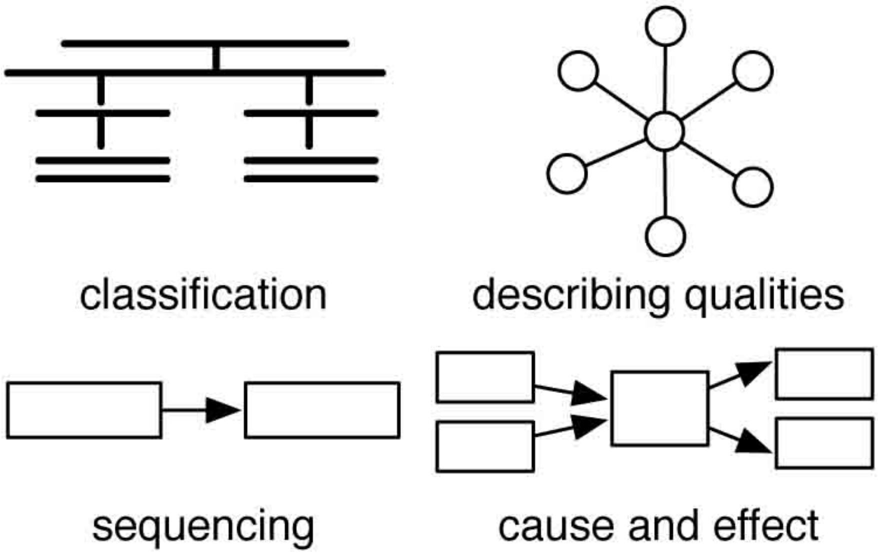


***Figure 5**: Some Thinking Maps graph types*

This quest of patterns in knowledge organisation is in high contrast with the information sketching approach taken by Mind Maps [14], where no formal criteria whatsoever are applied, and unconstrained brainstorming-type associations appear as nodes connected by branches (Fig. 6).

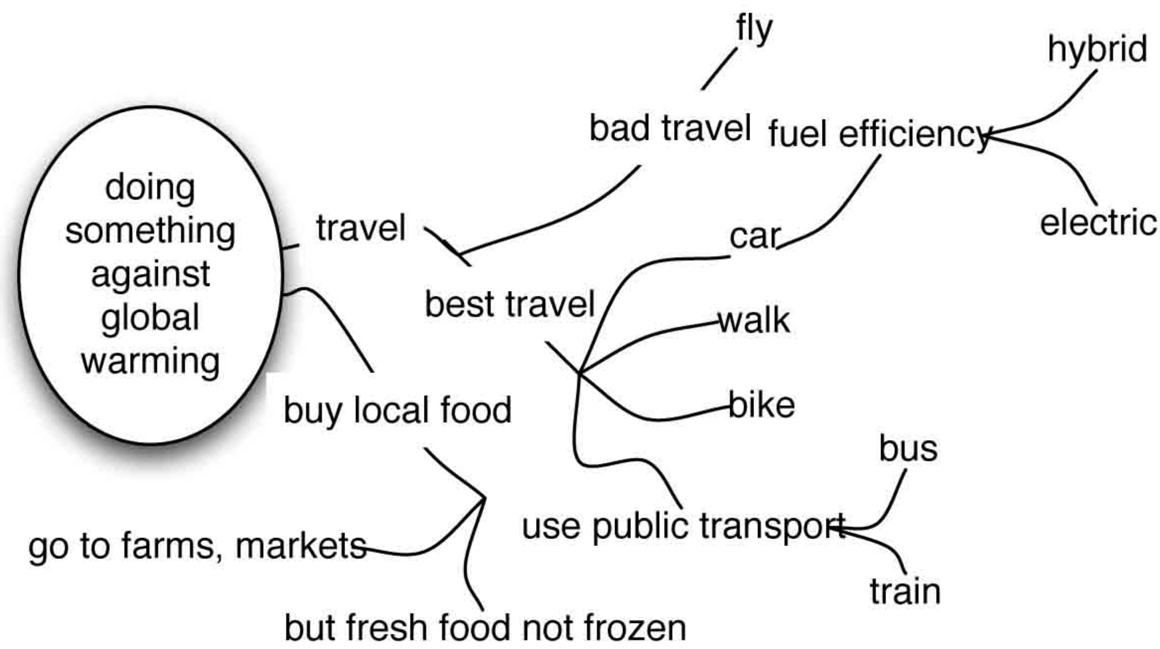


Figure 6: A fragment of a Mind Map

2.2. Addressable resource-connected knowledge graphs

Besides picturing a common understanding of the entities that populate some domain (their names, their properties and the relations holding between them), knowledge graphs designed to make web content accessible for semantic queries, include a distinguished feature, namely pointing at electronic addresses. This facility allows to identify the meaning of retrievable resources (webpages, for example) from the perspective of a previously defined conceptual structure. Semantic identification occurs when linking resources to a concept included in the conceptual structure, or binding resources via more complex relations (such as "has colour", which may connect a resource that describes "blue" to some other one describing "sky").

A webpage about someone called "John Smith" can, for instance, be connected to a webpage representing some organisation, say the "Happy Future Company" by means of an "Employment" relation, to express that John Smith is employed by this company. The above information may be relevant to a query of type "Who is John Smith's employer?" by using the resources already existing on the network. The major approaches tackling this problem are RDF-based ontologies and Topic Maps. They were both developed in the 90' and are now in care of different standardising organisations, the W3C and the ISO, respectively.

RDF

Diagrams are the canonical presentation of RDF expressions made available for reading by humans. RDF diagrams are triplets relating a subject-node (what is being talked about) to a property viewed as an attribute-value pair. The attribute is figured by an arc tying the subject-node to the value of the attribute (the "object" node in RDF terminology). An RDF equivalent of the sentence "The sky is blue" would thus turn "Sky" into a subject-node and interpret "is blue" as the "has colour: blue" property, where "has colour" acts as an attribute and "blue" as the value given to it.

Within an RDF diagram, ellipses represent a referent ("resource", in RDF vocabulary) by means of a URI (a computer-interpretable unique identifier), rectangles stand for some literal value, and arcs for a predicate that may pertain to a defined semantic datatype. Except for "blank" nodes, standing for what is being defined, every element of the triplet (excluding literals) is bound to a URI (usually an electronic address, but also a registered identifier, such as an ISBN number). Classes pre-defined in RDF or in compatible concept collections, such as FOAF (example: "Person"), as well as properties (see "is of type" [1], "has full name" [2] and "has mailbox" [3] in Figure X) are referred to by the corresponding URIs. Figure 7 shows three triplets, meaning that the described individual is a person [1], that the person's name is "Tanguy Durand" [2] and that the person's email address is "td@timgroup.fr" [3].

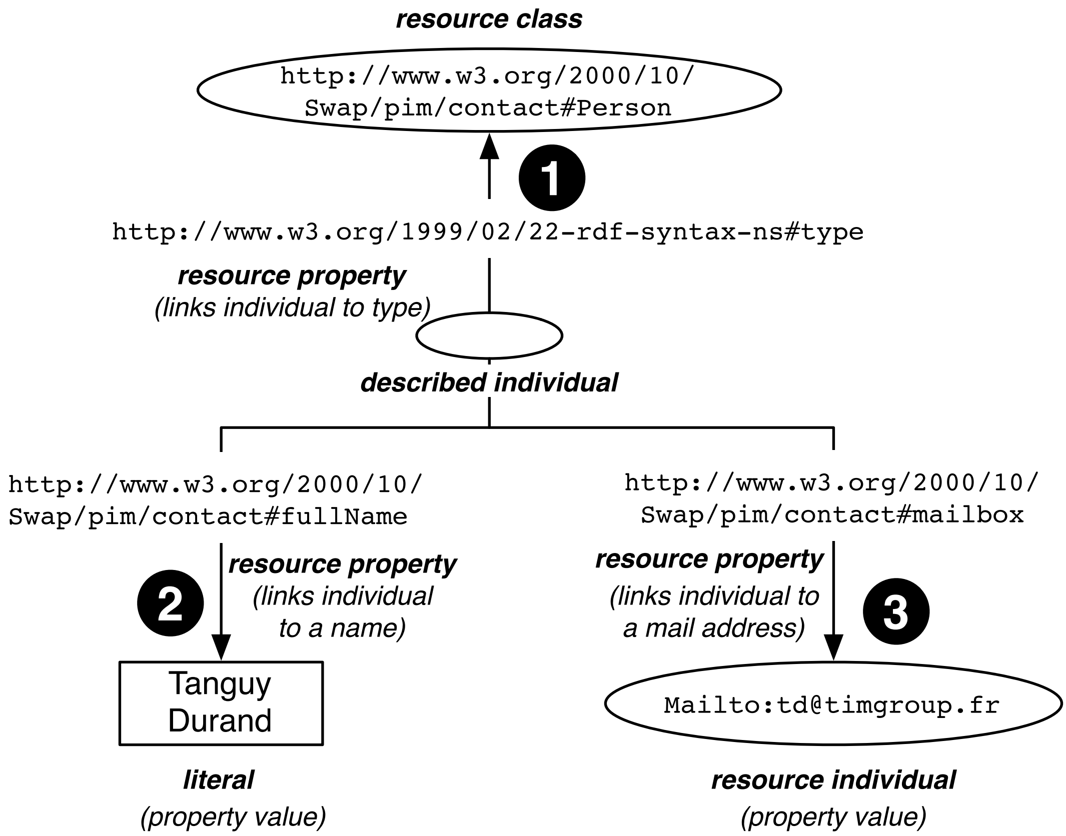


Figure 7: *RDF triplets*

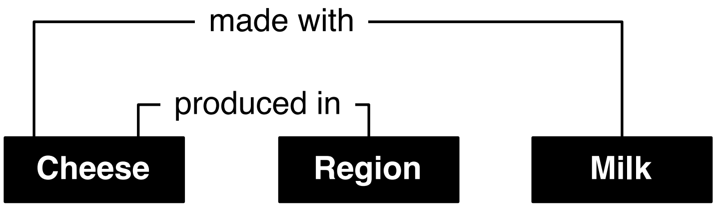
As can be easily noticed, RDF constructs translate into highly complex diagrams when considered from the point of view of standard human readability skills. The fact that nodes and arcs are presented as URIs (perfect as it may be for machine processing), is a considerable hindrance for a straightforward understanding of the graph's content. This follows from an architectural choice implemented by RDF, where the identity of an element (expressed by a URI) stands for the element itself.

Topic Maps

Although Topic Maps are structured as graphs, visual representations of them in the reference literature (standards, key articles), do not accommodate stable conventions, as do RDF diagrams. Nevertheless, the term "map" strongly suggests that its components can be given some kind of spatial distribution. The nature of the invoked space appears in the Topic Map literature as divided into two layers: the *knowledge* layer and the *information* layer.

The knowledge layer contains reified subjects, i.e. "Topics", which can be bundled into categories ("Topic Types"), a topic being an instance of a topic type ("France" is instance of "Country", for example). Topic types may undergo hierarchical embedding, which allows the "Artist" topic type to be a subclass of the "Person" topic type, given that every artist is a person. The knowledge layer also contains relations between topic types. Relations link topic types via rôles (the "Creation" relation would tie two topic types, such as "Artist" and "Work", further specifying that the rôle of an instance of "Artist" with respect to an instance of "Work" will be "created" and that the rôle of an instance of "Work" relatively to an instance of "Artist" will be "was created by").

As in ontology engineering the content of a described domain depends on what the analyst considers as relevant to his purpose, not only the objects but also the properties put forward to describe them are chosen according to goals. These goals are the "questions" the ontology is designed to answer. In our example (a toy Topic Map about French cheeses), the goal is to show which region a certain type of cheese comes from and what type of milk is used to make it. To illustrate how the topic map may be built step-by-step, figure 8 shows a first draft of the map containing the types needed: Association Types (produced in, made with) relating to Topic Types (Cheese, Milk, Region).



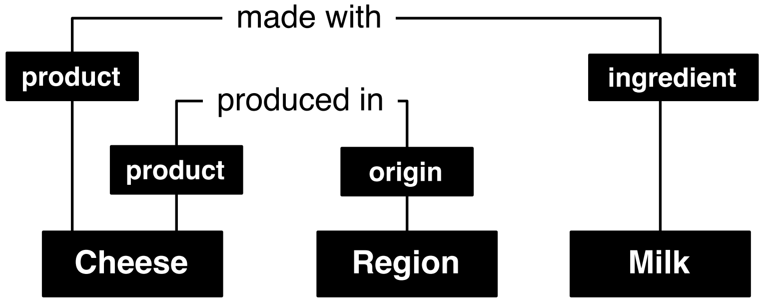
The goal of this conceptual architecture is to show *which region a certain type of cheese comes from* and *what type of milk is used to make it.* This first draft of the map contains the types needed: Association Types (*produced in*, *made with*) relating to Topic Types (*Cheese, Milk, Region)*.

***Figure 8**: Topic Map, first draft*

As within associations the concerned Topic Types play some rôle, the second step is to define the basic function of the topics in the associations involving them. It is handy to create rôles as types because the same rôle type can be part of different associations, as may be seen in figure 9.

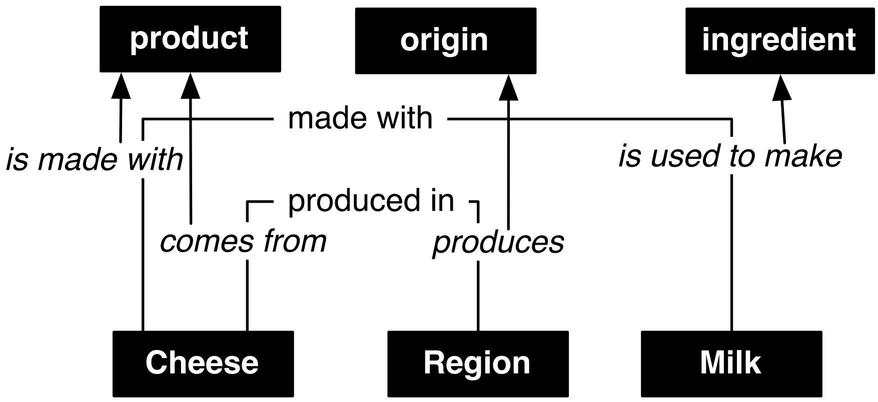
In our example, both associations share the "Product" rôle. To allow rôle sharing, the types are given instances (*is made with* and *comes from* are two instances of the *product* Rôle Type, see figure 10). To enhance readability, names of the instances should allow for a natural wording of the relation (such as "Cheese is made with milk" and "Milk is used to make cheese »).

Once the conceptual structure is defined, instances can be added to the Topic Types that provide arguments to the relations (figure 11, left). Describing these instances implies relating the appropriate ones, following the defined association scheme (figure 11, below).



Here Rôle Types are added, showing what the basic function of a topic is in a relation (*product*, *origin* and *ingredient*).

***Figure 9**: Topic Map, second draft*



Both associations share the "Product" rôle. To allow rôle sharing, the types are given instances (*is made with* and *comes from* are two instances of the *product* Rôle Type). To enhance readability, names of the instances should allow for a natural wording of the relation (such as "Cheese is made with milk" and "Milk is used to make cheese").

***Figure 10**: Topic Map, third draft*

The full picture of instantiated topic, association and rôle types appears in figure 12. This part of the Topic Map is known as the "knowledge" layer.

Every element in the knowledge layer can be documented (referred to resources). These resources can equally be typed ("article", "book", "file", etc. can be seen as resource *types*). The documentation of the knowledge layer builds the *information* layer of a topic map (Figure 12). At first glance, this twofold architecture just allows for a sound separation between the conceptual content of a Topic Map and the documentation of the content's building blocks, that is, a human-readable domain description enhanced with documentation. In fact, the interaction of both layers also adds meaning to resources in two ways.

Every element in the knowledge layer can be documented (referred to resources). These resources can equally be typed ("article", "book", "file", etc. can be seen as resource *types*). The documentation of the knowledge layer builds the *information* layer of a topic map (Figure 12).

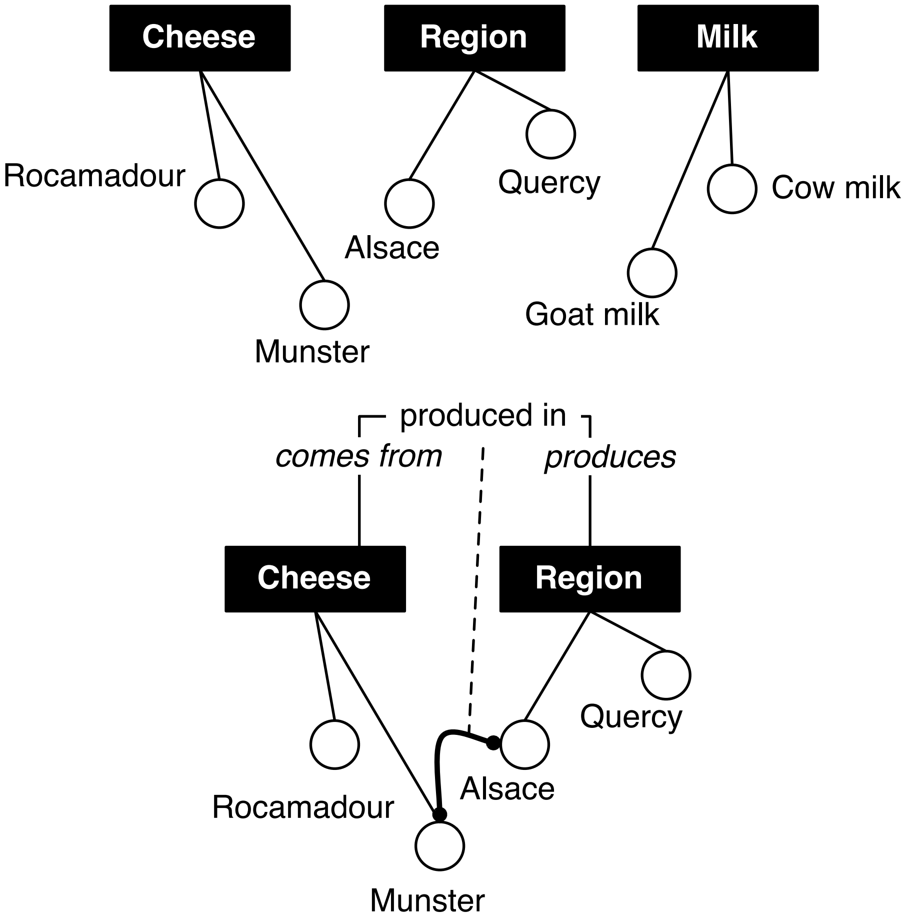


Figure 11*: Topic Map, Instances and connections*

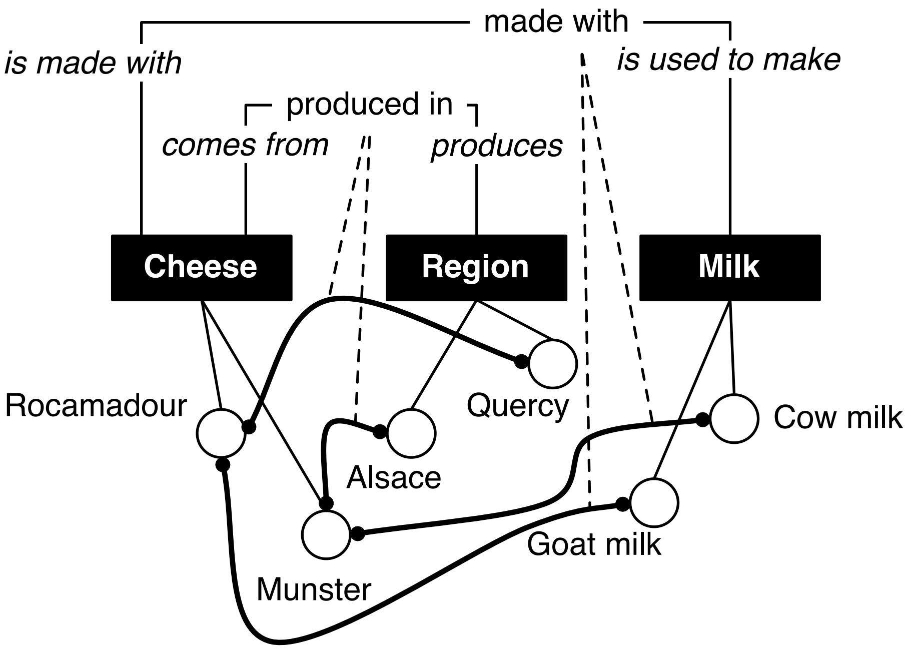


Figure 12*: Complete Topic Map (Information Layer)*

Firstly, connecting a resource with a Topic, unveils the semantic value of the resource's content with respect to a given domain model (the content of an encyclopaedia entry becoming the canonical definition of some given topic, for example). Secondly, asserting associations between topics (as are the "Alsace" region and the "Munster" cheese are in our example), allows to satisfy "complex" queries, such as obtaining information about what kind of products originate in Alsace or where the Munster cheese comes from. It is the knowledge layer that is here acting as a metadata superstructure, bringing up answers to complex queries as well as documentation background for the components it contains.

However, as searches of type "What kind of products originate in Alsace" are to be coded in a specific Topic Map query language, a graphical or form-based interface is needed in order to spare discomfort to end-users. This we have considered to be the basic requirement when setting up the system described below. Before going into the details of it, section 2.3 summarises some differences between the Topic Map and the RDF-based ontologies approach.

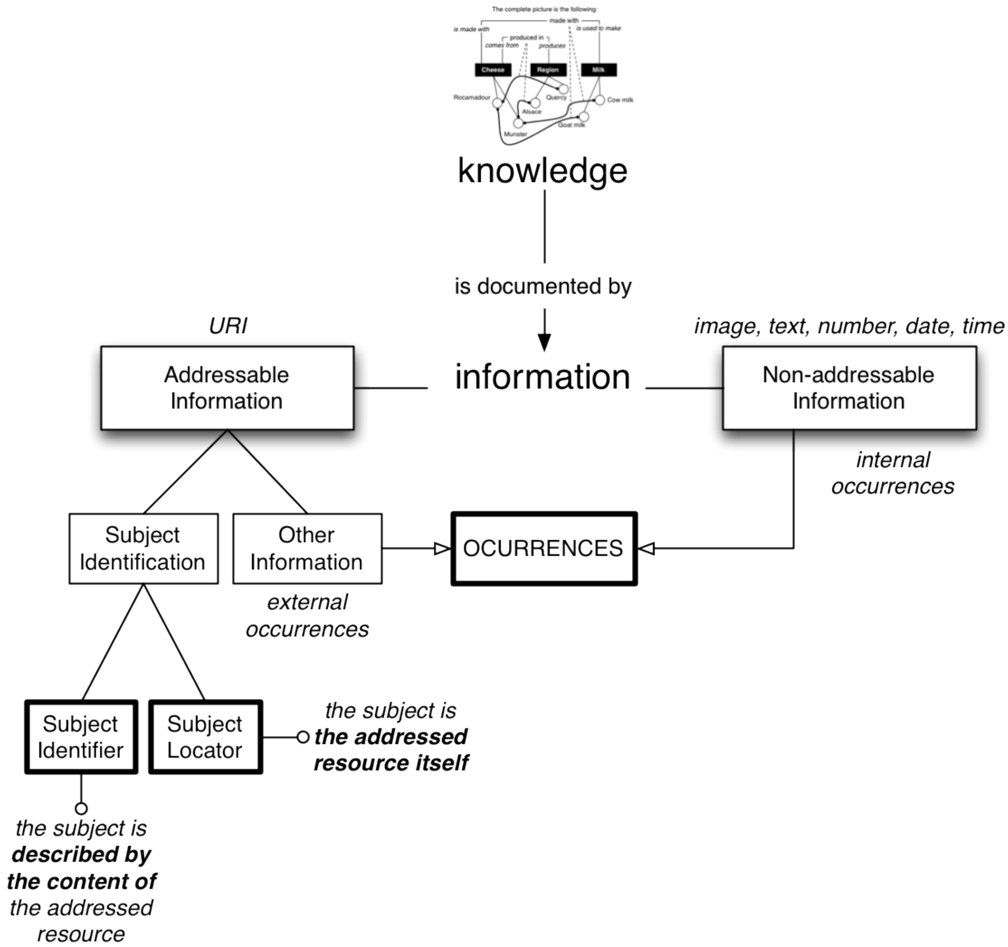


***Figure 13**: The Knowledge and the Information layers of a Topic Map*

2.3. A brief comparison between Topic Maps and RDF graphs

Table 1 shows some contrasts between RDF graphs and Topic Maps. Beyond terminological differences, the comparison underlines both the expressiveness and the flexibility of the latter. Moreover, as rendering of the knowledge layer of Topic Maps by graphical or form-based means appears as a reasonable challenge, but mainly because the autonomy of it allows for effortless understanding of the domain model it stands for, Topic Maps seem to provide a satisfactory solution for metadata-based management of wikis.

Table 1: RDF ontologies and Topic Maps

Domain of comparison	RDF-based ontologies	Topic Map-based ontologies
Referent (Represented Thing)	"Resource" (anyhing)	"Subject "(anything)
Referent Representation	"Node"	"Topic"
Name of assertions	"Statement"	"Topic Characteristics"
Content of assertions	"Properties" relating a "subject" (node) to an "object" (node)	"Names"
		"Occurrences"
		"Associations"
Naming	"Name" is a property of a node expressed in metalanguage	"Name" is the label of a topic
Rôle of URIs	Used to identify every node that is non-blank or literal and properties	Used to identify information resources relevant to a topic
Semantics of URIs	No way to know if the subject is the resource or its content	Different labeling for subject = resource and subject = resource's content
Graph capacity	triplet	n-ary
Rôle specification (in relations)	no	yes
Inbuilt hierarchical relations	none	supertype / subtype, type instance
Scoping support	no	yes
Universal typing facilities	no	yes
Architecture	Adherent knowledge and information layers	Non-adherent knowledge and information layers

3. Collaborative knowledge architectures

As defined in Wikipedia [16], a wiki is "a website that allows the easy creation and editing of any number of interlinked web pages via a web browser using a simplified markup language or a WYSIWYG text editor. Wikis are typically powered by wiki software and are often used to create collaborative websites, to power community websites, for personal note taking, in corporate intranets and in knowledge management systems."

We think of the content of a wiki as webpages. These pages can be considered as atoms or as an aggregation of page fragments that may be addressed separately. The semantic superstructure designed for selective retrieval of content elements, we consider to be the wiki's concept layer. As this superstructure is usually embedded in the wiki, the distinction between the content and the concept layer is here intended to be merely functional.

Notwithstanding, separating both layers, can extend the definition of a wiki beyond the borders of a website: the concept layer being on some particular address, its elements pointing to content distributed on pages that belong to different sites. Social bookmarking could this way qualify as a particular type of wiki implementation, and "folksonomy" as a perspective on the engineering of shared semantic superstructures.

Viewing layers as separate does not mean that structure (understood as dependencies between elements) is bound solely to the wiki's conceptual component (reflecting relations between concepts) or to the wiki as a whole (links connecting elements of both layers). As will be seen, the content layer may itself exhibit some kind of (navigational) structure, namely when content units (such as articles) point at each other, regardless of what their connections to the conceptual layer may be. In order to earn insight into existing design policies related to this issue, we will here discuss three different models of architecture, corresponding to the approaches taken by *Wikipedia*, *Del.icio.us* and *Fuzzzy* to promote the sharing of semantically structured resources.

Wikipedia

Zesch and Gurevych [17] describe the structure of Wikipedia as two connected graphs, the Wikipedia Category Graph (WCG) and the the Article Graph. The Article Graph represents the content of the encyclopedia by a set of related nodes, each one standing for an article. A node may point to any number of other ones (articles are "heavily linked", as the authors underline). Cycles are thus standard at the article level, whereas at the category level (where some exist [18]), they generate inconsistencies, due to the fact that the structure of categories is built as a (loose) taxonomy based on hyponymic and meronymic relations [17]. For this reason, articles cannot just be seen as instances of categories within a single graph.

As PehcevskiIn et al. [19] point out, the WCG, "[…] the sub-category relation is not always a subsumption relationship; for example, ‘‘Maps of Europe’’ is a sub-category of ‘‘Europe’’, but the two categories are not in an *is-a* relationship". Because links (i.e. relationships) are not qualified, but also because the same category can have more than one parent category, the WCG provides the facilities of an association network assisting heuristic search rather than those of a domain model allowing reliable inferences.

Fuzzzy

Fuzzzy (http://fuzzzy.com/) is a collaborative site hosting contents (video files, audio files and information about books) as well as bookmarks of webpages and feeds. Each one of these may be retrieved by their name but also by tags added to them. Tags are semantic annotations acting as labels that can be shared by different contents. Figure 14 shows the basic structure of Fuzzzy. Users may add content, add tags and also affect tags to content.

A distinguished feature of Fuzzzy tags is that they are Topic Map flavoured: a tag, which can be thought of as a Topic Type, can be related to another tag by a system-defined relation. The relation is equivalent to a Topic Map association baring two rôles, that can be linked to the participating tags via an association editor. When browsing content by tags, related tags appear list-wise under the description of the selected one. The relation of these to the selected one can be obtained by dragging the mouse over them.

However, the Fuzzzy implementation of Topic Maps is narrow. In the first place, associations are system-defined, and thus disallow tag (Topic Type) relations that fall outside of the range covered by the existing ones.

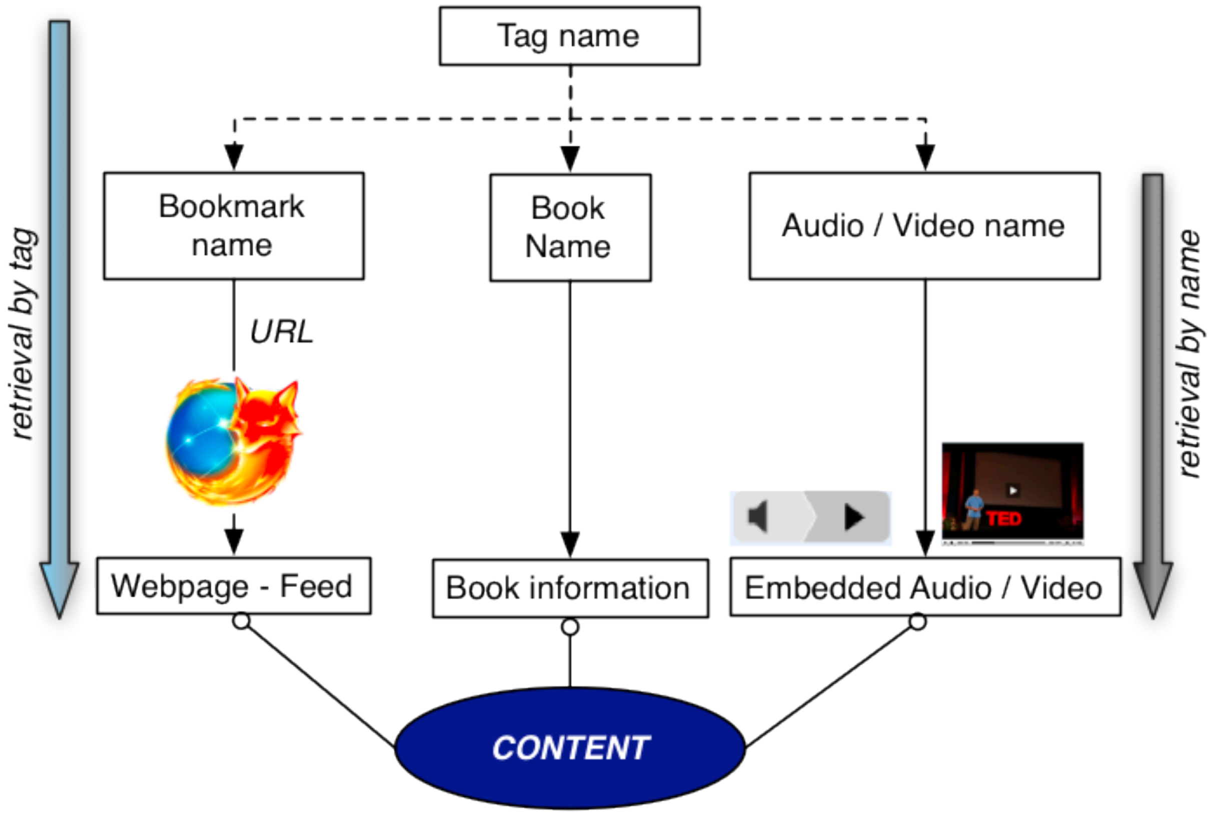


Figure 14: *Fuzzzy : an information tagging structure*

These are necessarily symmetric and accommodate exactly two rôles. The associations are only accessible by tag pairs (the ones playing a rôle in an association that remains itself invisible when browsing), which results in a very restricted view of the knowledge layer. Last but not least, the lack of graphical display of the knowledge layer provides a very fragmented view of its content.

References

[1] Novak, J. D. (1998). Learning, creating, and using knowledge: concept maps as facilitative tools in schools and corporations. Mahwah, NJ: Lawrence Erlbaum Associates.

[2] Ifenthaler, D.(2007). Relational, structural, and semantic analysis of graphical representations and concept maps. http://www.aect.org/pdf/proceedings07/2007I/07_23.pdf

[3] Sowa, J. - Semantic Networks - www.jfsowa.com/pubs/semnet.htm, Last Modified: 06/03/2006 03:52:13

[4] Øhrstrøm, P., Uckelman, S., Schärfe, H. (2007). Historical and Conceptual Foundation of Diagrammatical Ontology. In Priss, U., Polovina, S., Hill, R (Eds.): ICCS 2007, LNAI 4604, pp. 374–386, 2007. © Springer-Verlag Berlin Heidelberg 2007.

[5] Gurr, C. (2001). Aligning syntax and semantics in formalisations of visual languages. In Proceedings of the IEEE Symposia on Human-Centric Computing Languages and Environments (HCC'01) 0-7695-0474-4/01 © 2001 IEEE (04740060).

[6] Jill H. Larkin H., Simon, A. (1987). Why a Diagram is (Sometimes) Worth Ten ThousandWords. In Cognitive Science 11, 65-99 (1987).

[7] Greeno, J. (1994). Gibson's Affordances. In Psychological Review, Vol. 101, N° 2, pp. 236-342.

[8] Brachman,R., Schmolze, J. (1985). An Overview of the KL-ONE Knowledge Representation System. In Cognitive Science 9, pp. 171-216 (1985).

[9] Quillian, M. (1968). Semantic Memory, in M. Minsky (ed.), Semantic Information Processing, pp 227-270, MIT

[10] Collins, A., Loftus, E. (1975). A Spreading-Activation Theory of Semantic Processing. In *Psychological Review* Vol. 82, No. 6, 407-428 (1975).

[11] Cañas, J., Novak, J. (2006). Re-examining the Foundations for Effective Use of Concept Maps. In Concept Maps: Theory, Methodology, Technology. Proc. of the Second Int. Conference on Concept Mapping. San José, Costa Rica, 2006.

[12] O'Donnel, A., Dansereau, D., Hall, R. (2002). Knowledge Maps as Scaffolds for Cognitive Processing. In Educational Psychology Review, Vol. 14, No. 1, March 2002.

[13] Hyerle,D. An Introduction to Thinking Maps Research and Practice. Online http://www.thinkingfoundation.org/. Retrieved December 1 2009

[14] Buzan, T, Buzan, B. (1993). The Mind Map Book. London. BBC Books. ISBN 0-525-93904-0.

[15] Chen, P. (1976). The Entity-Relationship Model - Toward a Unified View of Data. ACM Transactions on Database Systems, 1(1):9-36, March 1976.

[16] http://en.wikipedia.org/wiki/Wiki

[17] Zesch , T., Gurevych, I. Analysis of the Wikipedia Category Graph for NLP Applications. https://www.tk.informatik.tu-darmstadt.de/fileadmin/user_upload/Group_UKP/publikationen/2007/hlt-textgraphs.pdf. Retrieved December 1 2009

[18] Lizorkin, D., Medelyan, O., Grineva, M. Analysis of Community Structure in Wikipedia (Poster). http://mint.typepad.com/mypapers/pp017-lizorkin.pdf. Retrieved December 1 2009.

[19] Pehcevski, J., Thom, J.A., Vercoustre, AM. et al. Inf Retrieval (2010) 13: 568. doi: 10.1007/s10791-009-9125-9

[20] Specia, L., Motta, E. Integrating Folksonomies with the Semantic Web. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.141.3540&rep=rep1&type=pdf. Retrieved December 1 2009.