Concept Maps: Theory, Methodology, Technology
Proc. of the First Int. Conference on Concept Mapping
Pamplona, Spain 2004

THREE POLES FRAMEWORK FOR CLASSIFICATION OF VISUALIZATIONS

Mathew Koshy, St. Petersburg State Polytechnical University, Russia
Email: matkoshy@hotmail.com

Abstract. As the importance of visualization grows day by day due to the massive knowledge, information and data generated by various sources like the World Wide Web, books etc. various tools and techniques have been designed and developed to handle complex visualizations. We propose a novel framework for classification of such tools and techniques with special emphasis on concept maps.

1 Introduction

One of the basic objectives of most research is to help mankind become more cognitive or to make cognitive artifacts. The former is a modest way of using technology. Visualization helps man to comprehend complexity, and to take intelligent decisions quickly. It is always observed that even while constructing such tools for visualization we encounter quite opposing forces like human comprehensibility versus machine processing. In the next section we propose a framework for classification which is based on these opposing forces, which in turn makes explicit the strength and weakness of the classified, in our case it being concept maps. We conclude with a simple strategy which we feel comes closer to striking a balance between these forces.

2 Three poles framework for Classification of visualization

This section describes the way of grouping different techniques and tools including visualization, based on the framework called the three poles of cognition (Koshy, 2004). This is essential since the terminology of cognition is itself a debate. Each of these poles are in orthogonal directions to each other like the axes of coordinate system and represents a way of thinking, defining and achieving cognition which are usually opposing to each other. The following is a brief sample space of the abstract “cognition”. Each element of the sample space “Cognition” is an ordered triple representing the tree poles. Each pole within an element is closely linked or related with the equivalent pole in another element. The following is a brief sample space. {(Readability, Expressivity, Computability), (Connectionist Machines, Symbolic Machines, Abstract Machines), (Information, Knowledge, Data), (Random, Incomplete, Uncertain), (function/graphs, logic, sets), (mapping, Semantics, Syntax/ Notation), (Interpretation, Reasoning, verification)…}. Cognition could be easily defined as the largest set containing all the above elements.

In the remaining section we would consider only the poles of (information, knowledge, data), (Readability, Expressivity, Computability) and (graph, logic, sets) because of their relevance to visualization, and see how to position concept map within this framework. It is very important to note that every entity classified has varying degree of each pole. For example every classified entity is a data, an information and knowledge in varying degree.

2.1 Classification based on poles of Information, Knowledge, Data

One of the major differences on knowledge and information is that information relies heavily on the concept of metadata whereas knowledge on the other hand concentrates on capturing propositions.
Edward Tufte, the world's leading thinker on using visualization for information design, describes two fundamental rules for visual display (Tufte, 2001):

1. Maximize the Data-Ink Ratio: Every drop of ink, or pixel on your screen, ought to be information bearing. Anything that appears simply for decoration should be removed.
2. Maximize Information Density: Tufte's research shows that given a choice, people prefer displays with more rather than less information. This may sound surprising, given the volume of complaints from people drowning in data.

Tufte also observes that old-fashioned maps are a superior means of applying these rules, as they convey more information per square unit of display area than any other presentation technique. Cartographic interfaces are based on these principles. Hyperbolic interfaces are an organized hierarchy of categories that opens and closes to expose information for browsing. As users click to the lower levels of the tree, the interface expands and/or rotates so the chosen item is centered and its links are more apparent. The ends of the tree link to the individual objects. Weighted spring interfaces are tree-like representations of non-numerical, contextually related objects and categories linked through the use of interconnected lines. They are similar in operation to hyperbolic trees however the linkages between levels in the tree are contextual as opposed to hierarchical; this allows users to see the association between objects. Finally the heat maps, also known as tree maps, Heat Maps are a map representation of a hyperbolic tree. They use color and size to represent metadata-rich numerical information. Heat Maps are most similar in appearance to cartographic interfaces and their presentation is very interactive and malleable, making Heat Maps excellent for identifying trends. However, their reliance on quantitative data places limits. (Antartica, 2003) It must be noted that the above principles for visualization of information designs is not applicable to visualization of knowledge design. Finally flowcharts serve as a great tool for visualization of the flow of data. The concept map summarizing this classification is given in fig. 1

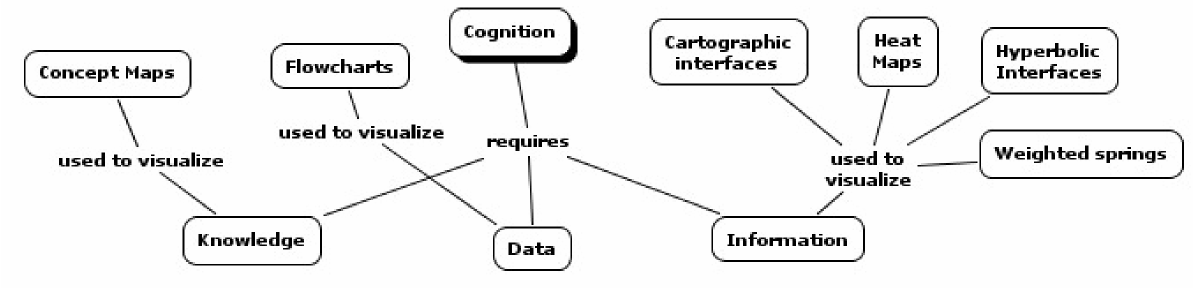

Figure 1. Cmap of the classification of Concept Maps in pole (Information, Knowledge, Data).

2.2 Classification based on poles of Readability, Expressivity, Computability

One of the key features of any knowledge representation based languages is expressivity. On the other hand graph based Languages provides greater readability and faster comprehension for mortal humans. If we consider machine cognition, we have to take into account the factor of computability too. Concept Maps which are basically focused on human cognition are both highly expressible and readable. A similar graph based representation is UML which are less expressive as the model element “relationship” is far restricted than concept map’s “linking word”. The UML metamodel is described in a combination of graphic notations, natural language, and formal language. The authors recognized that there are theoretical limits to what one can express about a metamodel using the metamodel itself. However, their experience suggests that this combination strikes a reasonable balance between expressiveness and readability (OMG, 2003) i.e. making it more readable. Ontological languages concentrates on a subclass of logic with well defined computable properties. The concept map summarizing this classification is given in fig. 2

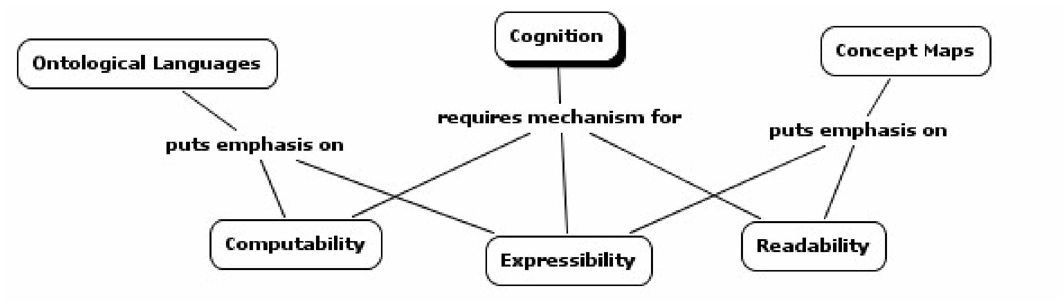

Figure 2 Cmap of the classification of Concept Maps in pole (Readability, Expressivity, Computability)

2.3 Classification based on poles of Graph, Logic and Set

Moshe Vardi describes graph theory, automata theory, and logic as the “holy trinity” of automated verification (Vardi, www). The pole of graph, logic, and set has very strong association with the pole of Readability, expressivity and computability. Linking words of concept maps can be considered as connectives of a higher order multi-sorted informal language called English (Richardson, 1995). It’s more expressive than the usual connectives of first order logic. On the other hand ontological languages are usually within the realm of first order logic. The concept map summarizing this classification is given in fig. 3

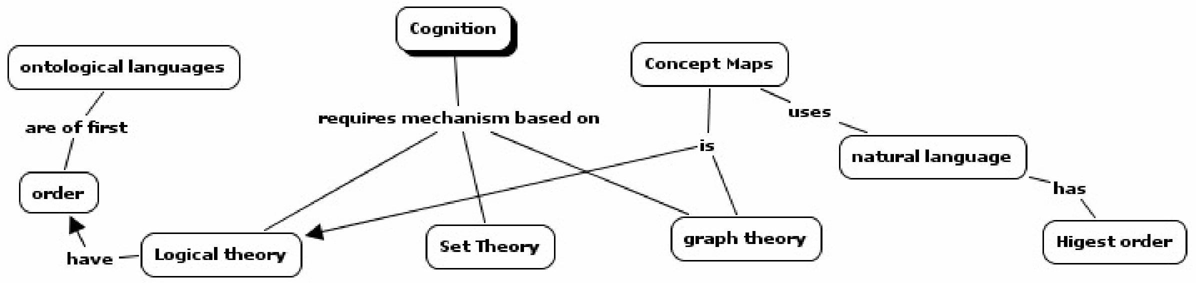

Figure 3 Cmap of the classification of Concept Maps in pole (graph, logic, set)

3 Advantage of the framework

One of the major advantages of this classification based on three pole framework is that it helps to discover the merits and demerits of concept maps. We can easily discover that concept map lacks strength in pole related with set theory, computability, and data oriented etc. we propose a simple strategy for concept map development that could tilt it towards this pole of cognition.

We very well know that flow charts have a start node and a terminal node and an arrow showing the flow of control. The first step would be to incorporate this idea with a modification into concept maps. In our case we have a start node and a terminal relation indicated by dashed lines. Start nodes have curved corners where as intermediate nodes have straight corners. We start at any "start node" and end at any node right after the terminal relation. As a demonstrative example we create the modified concept map on UML class diagram. It would be more convenient to term them as flow CMaps.

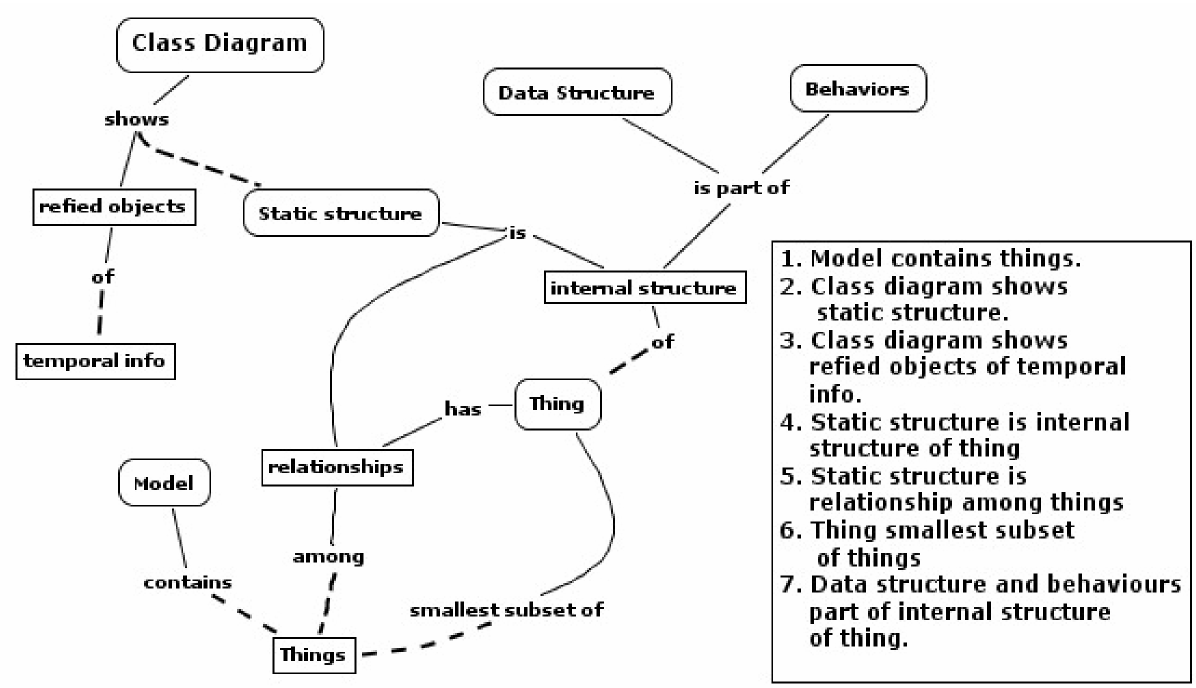

Figure 4 Modified Concept map (flow CMap) of UML class diagram along with computable sentences

This we feel that flow CMaps is more computable than concept maps/ flowchart hybrids which basically creates a node called procedures and uses numerals as linking words in order to generate an order on concepts. A few examples of such hybrids are available at (IHMC, www). An abstract concept map/ flowchart hybrid is shown on the left side of figure 5 for quick reference. The basic difference between flow CMaps and hybrid CMaps is that we are able to capture the flow of knowledge rather than just a flow of data. Another advantage is that we can have multiple flows within a single concept map. But the flow CMaps is not a replacement for Concept maps. It loses its visual elegance and simplicity. In order to counter balance this force and as our second step we propose a two layer architecture for CMaps. The traditional CMap should be used for visualization purpose and flow CMap be used for construction purpose. One of the immediate benefits of making concept maps more computable is that we could generate new CMaps on fly from a knowledge soup based on user queries. Suppose we search on the knowledge soup called UML about "what is a class diagram", and then the tool would display the appropriate CMap as shown in the right part of Fig. 5 hiding the irrelevant concepts.

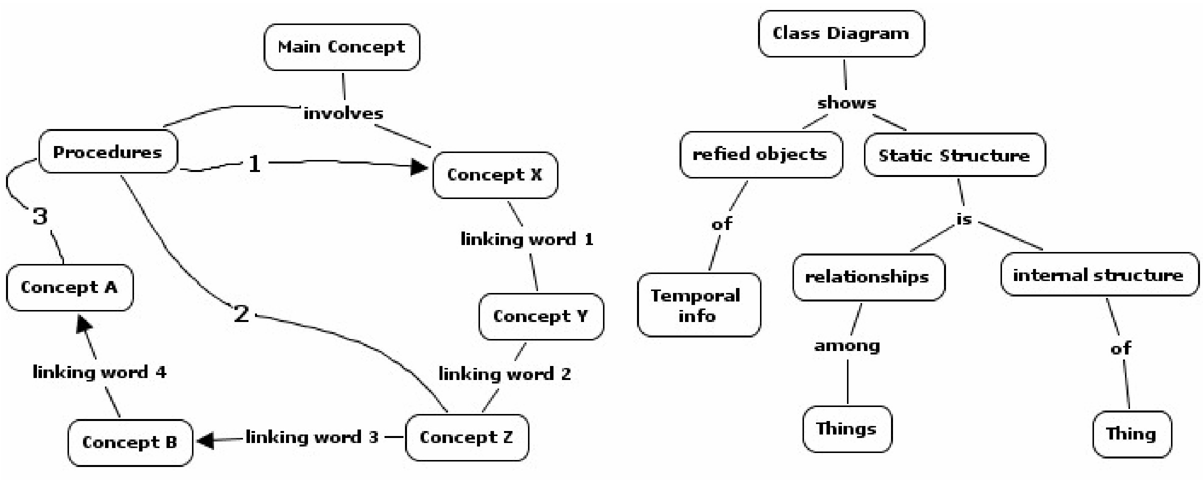

Figure 5. Left: abstract concept map of hybrid cmap, Right: result of query of "what is a class diagram?"

4 Summary

The three pole framework based classification provides a deeper insight into the weakness of concept maps by showing that it is very weakly computable and its strengths of readability and visualization of knowledge. We then try to strike a balance by incorporating the modified idea of flowcharts by creating a start node and a terminal relation. This gives rise to negative force against the elegance and simplicity of CMaps. This is counter balanced by proposing two layered concept map architecture by having CMaps exclusively for visualization purpose and flow Cmaps for construction and designing of CMaps.

5 Acknowledgements

This work was supported by personal grants from Mrs. Kong Yanfang and Mr. Daniel Koshy. I thank Prof. Lev Stankevich and the two anonymous reviewers for their critical review of previous version of the manuscript

6 References

Antartica (2003). *A white paper on information visualization*, http://www.antartica.net

IHMC (www). *Hybrid concept map/ procedure on building a Concept map*. Institute for Human and Machine Cognition. http://cmap.ihmc.us/Documentation/

Koshy M. (2004). *Situation Calculus based formalism for dynamical system modeling and its application to informalism in software engineering (UML)*, master's thesis, St. Petersburg State Polytechnical University

OMG (2003). *OMG Unified Modeling Language Specification v1.5*, (pp. 2-4), Object Management Group

Richardson D. (1995). *logic language Formalism Informalism*, Daniel Richardson, International Thomson Computer Press

Tufte E.R. (2001). *The Visual Display of Quantitative Information*, second edition. Graphics Press

Vardi M, (www) *seminar abstracts at Centre for Logic, Language and Computation*, university of Wellington. http://www.cllc.vuw.ac.nz/