Flexible Narrative Representations:

Bridging the gap between Formal Models and Informal Representations

Karin Breitman
PUC-Rio
Marquês de São Vicente 225
Rio de Janeiro -Brazil 22453-900
+55 21 3527-1500 ext 4360
karin@inf.puc-rio.br

Oscar Pastor
Universidad Politécnica Valencia
Camino de Vera s/n
Valencia – Spain - 46022
+34 963877210
opastor@dsic.upv.es

Simone Barbosa
PUC-Rio
Marquês de São Vicente 225
Rio de Janeiro -Brazil 22453-900
+55 21 3527-1500 ext 4353
sim@inf.puc-rio.br

ABSTRACT

As software engineers we were trained to use conceptual models to guide our activity. The more formal the better, for precise semantics secures clear interpretation, and the reduction of ambiguity. We are definitively believers. We have so much confidence in the models we build, that we automatically generate code from them (MDD). However, two decades and a reasonable amount experience in developing software, provided us with enough empirical evidence to conclude that there must be something missing. No matter how much friendly or intuitive the models used, we never get them right the first time, and end up having to resort to alternative "explanation" aids: drawings, mockups, mind maps, even live enactment.

In the Literary Mind, Mark Turner argues that the mind of stories and parables is not peripheral, but basic to thought [41]. Story is the central principle of our experience and knowledge, first to other kinds of thoughts. Story and metaphor are fundamental to help us understand everything in our experience, from getting organized in the morning to writing papers, such as this one.

In the last few years we have been actively investigating the use of similarity, analogy and metaphor to address key problems related to information systems design. We largely experimented with use of analogy and metaphor to support conceptual design. We used the notion of similarity to expand query results, and the notion of analogy to reformulate queries defined for one application domain into equivalent queries over a different domain. We also obtained good results using the notion of object similarity in matching conceptual schemas.

We boldly argue that narrative is the missing link, i.e., the instrument in which to bridge the dichotomy between formal modeling tools and more informal, free-form approaches. The rest of this paper is divided as follows. In section 1 we briefly describe our background experience in conceptual modeling. In section 2 we summarize our work using similarity, analogy and metaphor in information systems. In section 3 we discuss the use of narrative representations in Software Engineering, in section 4 we discuss our vision for a flexible narrative representation.

Categories and Subject Descriptors

D.2.2 [**Software Engineering**]: Design Tools and Techniques – *Object-oriented design methods, Petri nets, State diagrams.*

D.2.10 [**Software Engineering**]: Requirements/Specification – *Representation.*

General Terms

Design, Human Factors, Languages

Keywords

Requirements Engineering, Conceptual Modeling, UML, Use Cases, scenarios, annotation, MDD

Acknowledgement

This publication was made possible by grants number [E-26/170028/2008] from FAPERJ [557.128/2009-9] from CNPq, at the Brazilian Web Science Institute.

1. BACKGROUND EXPERIENCE

Our research focuses in modeling complex problems that can be solved computationally, i.e., identifying the essential properties and abstractions necessary to adequately represent a problem. Given that in most cases, more than one conceptual model is viable, we're also interested finding mappings between different representations. The depth of our research comes from the fact that we have been working on these topics for almost two decades. Width is given by the number of areas in which we have been experimenting with, summarized as follows:

In requirements engineering we have experimented with narrative models. Also known as scenarios, they generalize story type descriptions of situations that we want systems to deal with [10, 19]. They exist in several flavours, the most popular ones are UML's Use Cases [28] and Agile Method's user stories [5]. We have practical experience with information systems for laboratorial, hospital and sensor control systems

Our work in semantic web focuses in using Ontology as the means to formalize a domain theory, thus providing a precise way in which to refer to objects and their relationships. We have practical experience developing spacecraft, geo referenced, emergency response, automatic annotation and cultural heritage management applications [7, 12, 32].

In MDD by promoting a schema-centric development approach, based on the automatic translation of formal models into working code. We have large practical experience in model execution, having developed and made available an open source tool framework [35, 37, 38].

In database systems we are exploring schema integration, using a combination of structural, semantic and instance based

approaches. We have practical experience in autonomic system projects, more precisely in the application of the schema matching techniques in the alignment of heterogeneous ITIL-based representations [6, 13, 31].

In ubiquitous computing we are investigating heterogeneous mobile device model integration to provide interactive and collaborative smart spaces. We have practical experience in the development of projects with industry, MS active classrooms, and within the CAMPUS project framework with LIP6 - University of Paris VI (UPMC) [9, 42].

In cloud computing we are proposing conceptual model abstractions to help integrate cloud features into the early stages of software design. More precisely, we intent to define a set of abstractions that can be integrated with "more traditional" functional and non-functional requirements, in producing robust models that can be automatically translated into code (traditional MDD approach) [8,].

In the next section we provide more detail in the background work that is more central to the present discussion.

2. ANALOGY, SIMILARITY AND METAPHOR

Metaphor is not merely a rhetorical device, characteristic of language alone. Lakoff and Johnson [29] argue that “the human conceptual system is fundamentally metaphorical in nature. The essence of metaphor is understanding and experiencing one kind of thing in terms of another.” Holyoak and Thagard [27] argue that “*metaphor uses the same mental processes as analogical thinking ... a metaphor is understood by finding an analogy mapping between the target domain (the topic of the metaphor) and the source domain. The degree to which an analogy is viewed as metaphorical will tend to increase the more remote the target and source domains are from each other*.”

In our research we claim that analogy mappings facilitate conceptual modeling by allowing the designer to reinterpret fragments of familiar conceptual models in other contexts. Specifically, we proposed a discipline for conceptual schema design, and Semantic Web ontologies as well, that we call conceptual modeling by analogy and metaphor. The discipline is based on two simple ideas. First, a team of expert conceptual designers would build a standard repository of source conceptual models that cover commonly found conceptual design patterns and that are expressed in familiar terms. The source conceptual models will naturally contain fully formalized integrity constraints, as defined by the conceptual design experts. Second, naïve designers would then create new target conceptual models in other domains by defining analogy mappings with the source conceptual models in the repository. The target models will then borrow the structure and the integrity constraints from the source models by analogy – essentially a combination of a straightforward renaming process with consistency checking. The design discipline would then consist of the gradual expansion of conceptual models for specific domains basically by repeatedly defining analogy mappings. We also extend this discipline into a five-step process that takes four spaces into consideration – the source, target, generic and blended spaces – as proposed elsewhere for widely different areas [].

We applied the notions of similarity and analogy to investigate three problems related to accessing data stored in a database. This first problem, called frame classification, consists of determining to which class a given entity instance belongs. To address this problem, we introduce a frame representation for instances and classes, organized in a specialization/generalization hierarchy, and

define a classification process based on a similarity criterion that takes into account the known property values an instance has, as well as how classes are structured. The second problem, called query by similarity, consists of finding instances that best match some arbitrary search frame, that is, finding instances that are similar to the one desired. As a consequence of the discussion about classification, we observe that the closeness criterion employed to match instance frames against class frames can be modified to measure similarity with respect to arbitrary frames, thus extending the power to perform queries over the available information. The third problem, called query by analogy, addresses how queries defined for one application domain can be reformulated as equivalent queries over a different domain .

We followed a similar strategy to reuse operation (or service) definitions in the same domain and across different domains. We explore the concept of similarity to organize a library of operations and predefined operation compositions, that we call plots, and to help reuse such objects in the same domain. By contrast, we apply the concept of analogy to reuse these objects across different domains.

We treated the construction of plots as a plan generation process. A plan generator should be able to align the plot events in a coherent sequence in view of objectives, whenever possible coming up with more than one plot, so as to provide alternative ways to reach the objectives. But plots are often more attractive when unplanned shifts are allowed to occur. This is arranged for in our proposal through the limited power given to users to interfere with the planner, causing certain discontinuities in the context, particularly concerning changes in the feelings and beliefs of certain characters. Finally, one should have the possibility to obtain from the planner a more detailed account of the events, by having them expanded into smaller grain actions.

It turned out that the need to consider these four notions – coherence, alternatives, transgressive shifts, details – informally prescribed in the above paragraph as desirable for any effective plot composition process, brings to mind four different types of relations between events: syntagmatic, paradigmatic, antithetic and meronymic, which in turn are associated with the so called four major tropes of semiotic research, namely metonymy, metaphor, irony and synecdoque.

We also applied similarity to investigate schema matching, a fundamental issue in many database applications, such as query mediation and data warehousing. A reasonable approach to schema matching, sometimes called extensional, instance-based or semantic, is to detect how the same real world objects are represented in different databases and to use the information thus obtained to match the export schemas [2, 3, 11, 24]. We are currently preparing a compilation of these results, to be published in a book entitled Analogy and Metaphor in Information Technology by Springer Verlag, Fall 2010.

3. NARRATIVE REPRESENTATIONS IN SOFTWARE ENGINEERING

In Software Engineering practice the closest representation we have for stories are the so called scenario-based representations, e.g., user stories, use cases, semi-structured scenarios [30, 16] and formal scenarios. Scenario-based representations are very effective tools to promote communication during the development process, in particular to promote validation with users. They are useful for describing functional system behavior, in the users' language, as opposed to using technical abstractions. Most representations, however, are not flexible enough to let users decide the granularity in which they want to capture information, i.e., they are either too cumbersome or too lightweight.

Scenarios are usually organized in semi-structured, template-like fashion. Most information is provided as text. It is also important to note that most representations do not work very well in cases where there is missing, or incomplete information.

We have been recently exploring the notion of plots as a form of narrative, and its applications in information systems. Literary research addresses narratives at successive levels. The most basic level, the fabula, is defined as "a series of logically and chronologically related events that are caused or experienced by actors" [1]. Intuitively, plots are the stories [40, 41] that happen in the underlying mini-world and, as a result, produce state-changes in its database representation. More precisely, an event represents the result of the execution of some domain-oriented operation by an authorized agent, and a plot is a partially ordered set of events.

Plot analysis is a rich source of knowledge about the agents' behavior when accessing data stored in the database. It relies on (logical) database logs, also called audit trails, which register the actions of individual agents. A trivial example of a log is a bank account statement, which records the sequence of actions executed against the account. A second example comes from storytelling engines, such as LOGTELL [14]; which model the world as a database and are based on a set of pre-defined actions and plots [15]. A log in this case is the trace of events generated by composing a story interactively. In the context of an emergency response information system [Van de Walle2007] a log registers the actions taken when handling an emergency, or during a training exercise [12].

In the next section we discuss the need for an alternative way in which to capture narratives, as neither the existing scenario-based nor the representation for plots offer the necessary levels of flexibility.

4. FLEXIBLE NARRATIVE REPRESENTATIONS: OUR VISION

We are definitively convinced that the use of narrative is the solution to bridge the dichotomy between existing formal models and informal flexible tools. A quick reflection on our day-to-day practice provides very good examples: we were tempted to include a figure of our argument, without reference in the text, only to prove the point that there is no such thing as a self-explanatory model; we must always provide a description in the text, even if an obvious one. The same is true for mathematical formulas, whose interpretation is precise by definition, nevertheless authors always provide an "intuition".

We must be careful, however, not to "through the baby out with the water". Existing formal representations are quite useful, and have improved Software Engineering practice a great deal. What we propose is an enhancement, an additional layer that will

provide more semantics and, more importantly, a link with additional information that will help us better understand requirements. Our idea is to provide a means in which to annotate existing formal representations, create stories that describe, explain these models, by linking to informal pieces of information. We envision several practical areas of application:

- MDD-based tools use a set of conceptual primitives that are fixed by some formal specification language. UML scenario type narratives could be used to "describe" such models by combining text with graphics (other UML representations, more informal ones), providing flexible ways in which to view models during system development (a great challenge to the success of the MDD approach [22]), where formal and informal aspects are intertwined [19, 26, 34, 35, 38].

- Biogenomic databases and workflow tools should include textual annotations intended to help to identify common semantics of different data representations. This particular application will facilitate the identification and matching of similar sequences and experimental procedures, promoting tool interoperability and facilitating reuse [36]

- Narrative Web pages should also include annotations elaborated with the aid of a formal ontology, to help distill the information in a way to build a Web ok Knowledge. [21]

- Historical databases could potentially benefit from annotations on data provenance and rationale. It is often the case that measurement techniques evolve as technology improves, resulting in more precision and the inclusion of new attributes. Projects that span long periods of time, e.g., Landsat and economic index collections are good examples [17,18, 23, 25, 31].

- Currently available tools for robot operation and spacecraft integration and testing that start with a formal model of a system and produce a provably equivalent implementation are valuable, but not sufficient. The "gap" that such tools leave unfilled is that their formal models cannot be proven to be equivalent to the system requirements as originated by the customer [33, 39]. The proposed approach may provide a working solution for a class of systems whose behavior can be described as a finite (but significant) set of narratives.

To achieve this goal, we must be much more accommodating of the format and of the types of information used in the narratives, than we have ever been (forms and structured text will no longer do). We have to be flexible enough to accept, mix, and match different bits of information such as pictures, mind maps, partial descriptions (the look-and-feel of x, the way of execution process w in tool z), videos, voice annotations, web pages, etc.

Granted that some of the pieces will be of little computational use, vague and/or incomplete. To that we reply as follows: We remark that a central question in Philosophy of Language is weather ambiguity belongs to Language or it is part of the world. Similarly, we question weather informality is in the representation, or is part of the World. Software engineering researchers have been tacking the first possibility by proposing representation languages that are more powerful, expressive and proved useful. If the second, then the elimination of informality constitutes a mis-representation of the World. Perhaps this is our chance to stand corrected.

5. REFERENCES

[1] Bal, M. Narratology - Introduction to the Theory of Narrative. U. Toronto Press, 2002.

[2] Barbosa, S.D.J.; Breitman, K.K.; Furtado, A.L.; Casanova, M.A. "Similarity and Analogy over Application Domains". In: Proc. XXII SBBD. João Pessoa, Brazil, Oct. 2007.

[3] Barbosa, S.D.J.; de Souza, C.S. "Extending software through metaphors and metonymies". Knowledge-Based Systems, 14, 2001.

[4] Batini, C.; Ceri, S.; Navathe, S. Conceptual Design – an Entity-Relationship Approach. Benjamin Cummings, 1992.

[5] Beck, Kent "Embracing Change with Extreme Programming". Computer 32 (10): 70–77, 1999.

[6] Breitman, K.; Brauner, D.; Casanova, M.; Milidiú, R. ; Gazola, A.; Perazolo, M. Instance Based Ontology Mapping. In: Proceedings of the Fifth IEEE Workshop on Engineering of Autonomic and Autonomous Systems. Los Alamitos : IEEE Computer Society Press, v. 1. p. 67-74, 2008.

[7] Breitman, K.; Casanova, M.; Truszkowski, W. - Semantic Web: Concepts, Technologies and Applications - NASA Systems and Software Engineering Series. 1. ed. London: Springer Verlag London, 2007.

[8] Breitman, K.; Endler, M.; Azambuja, M.; Pereira, R. - When TV dies, will it go to the Cloud? – IEEE Computer - vol. 43 no. 4 - April 2010, to appear

[9] Breitman, K.; Hinchey, M. - The use of Formal Ontology to Specify Context in Ubiquitous Computing. In: Third International Symposium on Leveraging Applications of Formal Methods, Verification and Validation, ISoLA08. Communications in Computer and Information Science , Vol. 17 - Springer Verlag, 2008. v. 17. p. 561-571, 2008

[10] Breitman, K.; Leite, J.; Berry, D. - Supporting scenario evolution. Requirements Engineering Journal, London, v. 10, no. 2, p. 112-131, 2005.

[11] Breitman, K.K.; Barbosa, S.D.J.; Casanova, M.A.; Furtado, A.L. "Conceptual modeling by analogy and metaphor". In: Proc. CIKM 2007. Lisbon, Portugal. Nov. 2007.

[12] Carvalho, M.T.; Freire, J.; Casanova, M.A. "The Architecture of an Emergency Plan Deployment System". In: Proc. III Workshop Brasileiro de GeoInformática, Rio de Janeiro, Brasil, pp. 19-26, Oct. 2001.

[13] Casanova, M.A.: Breitman, K.; Brauner, D.; Marins, A. - Database Conceptual Schema Matching. Computer (Long Beach), v. 40, p. 102-104, 2007 .

[14] Ciarlini, A.E.M.; Furtado, A.L. "Understanding and Simulating Narratives in the Context of Information Systems". In: Proc. 21st. International Conference on Conceptual Modeling, Tampere, Finland, Oct. 2002.

[15] Ciarlini, A.E.M.; Veloso, P.A.S.; Furtado, A.L. "A Formal Framework for Modelling at the Behavioral Level". In: Proc. 10th European-Japanese Conference on Information Modelling and Knowledge Bases, 2000.

[16] Cockburn, A. - Writing Effective Use Cases. Addison-Wesley, 2001.

[17] Cox, S.; Daisey, P.; Lake, R.; Portele, C.; Whitside, A. (Eds.) (OpenGIS® Geography Markup Language (GML) Implementation Specification. Open Geospatial Consortium, Inc, 2003.

[18] Di, L.; Yang, W.; Deng, M.; Deng, M.; McDonald, K. (2001) The prototype NASA HDF-EOS Web GIS Software Suite (NWGISS). In: Proceedings of the NASA Earth Science Technologies Conference. Greenbelt, Maryland. August 28-30, (CD-ROM, 4pp). Available at: http://laits.gmu.edu/Papers/NWGISS.htm.

[19] E. Insfrán, O. Pastor and R. Wieringa - Requirements Engineering-Based Conceptual Modelling - Requirements Engineering Journal - Springer Verlag - Vol 7, No. 2, 61-71, 2002.

[20] Egenhofer, M.J. - Toward the semantic geospatial Web. In: Proceedings of the ACM GIS'02, 2002.

[21] Embley, D. W., Liddle, S. W., Lonsdale, D., Nagy, G., Tijerino, Y., Clawson, R., Crabtree, J., Ding, Y., Jha, P., Lian, Z., Lynn, S., Padmanabhan, R. K., Peters, J., Tao, C., Watts, R., Woodbury, C., and Zitzelberger, A. - A Conceptual-Model-Based Computational Alembic for a Web of Knowledge. In Proceedings of the 27th international Conference on Conceptual Modeling (Barcelona, Spain, October 20 - 24, 2008). Q. Li, S. Spaccapietra, E. Yu, and A. Olivé, Eds. Lecture Notes In Computer Science, vol. 5231. Springer-Verlag, Berlin, Heidelberg, 532-533, 2008.

[22] France, R. and Rumpe, B. 2007. Model-driven Development of Complex Software: A Research Roadmap. In 2007 Future of Software Engineering. International Conference on Software Engineering. IEEE Computer Society, Washington, DC, 37-54, 2007.

[23] Fundação Getúlio Vargas - http://portalibre.fgv.br/

[24] Furtado, A.; Casanova, M.; Barbosa, S.; Breitman, K. - A Frame Manipulation Algebra for ER Logical Stage Modelling. In: 28th International Conference on Conceptual Modeling, 2009.

[25] Gallup institute - http://www.gallup.com/Home.aspx

[26] Giovanni Giachetti, Beatriz Marín, Oscar Pastor - Using UML as a Domain-Specific Modeling Language: A Proposal for Automatic Generation of UML Profiles. In Proceedings of the Advanced Information Systems Engineering, 21st International Conference, CAiSE 2009, 110-124, 2009.

[27] Holyoak, K.; Thagard, P. Mental Leaps. The MIT Press, 1996.

[28] Ivar Jacobson Object-Oriented Software Engineering. Addison Wesley Professional, 1992.

[29] Lakoff, G.; Johnson, M. Metaphors We Live By. U. Chicago Press, 1980.

[30] Leite, J.C.; Rossi, G.; Balaguer, F.; Maiorana, V.; Kaplan, G.; Hadad, G.; Oliveros, A. - "Enhancing a Requirements Baseline with Scenarios," Requirements Engineering, IEEE International Conference on, p. 44, Third IEEE International Symposium on Requirements Engineering (RE'97), 1997.

[31] Leme, L.A.; Casanova, M.A. - Breitman, K.; Furtado, A. - Instance-based OWL Schema Matching. Proceedings of the 11th International Conference on Enterprise Information Systems. Heidelberg : Springer Verlag, v. 1. p. 235-245, 2009.

[32] Leme, Luiz André P. ; Brauner, Daniela F. ; BREITMAN, K. K. ; Casanova, Marco A. ; Gazola, Alexandre . Matching object catalogues. Innovations in Systems and Software Engineering, v. 4, p. 315-328, 2008.

[33] Michael G. Hinchey, James L. Rash and Christopher A. Rouff, Requirements to Design to Code: Towards a Fully Formal Development Method, NASA GSFC Technical Monograph TM 2005 212744, January 2005.

[34] Oliveira, K.; Breitman, K.; Oliveira, T.; A Flexible Strategy-Based Model Comparison Approach: Bridging the Syntactic and Semantic Gap. Journal of Universal Computer Science, v. 15, p. 2225-2253, 2009.

[35] Pastor O., Gomez J., Insfran E., Pelechano V. - The OO-Method approach for information systems modeling: From object-oriented conceptual modeling to automated programming - Information Systems, 26 (7), pp. 507-534, 2001.

[36] Pastor, O. - Conceptual Modeling Meets the Human Genome - In Proceedings of the 27th international Conference on Conceptual Modeling (Barcelona, Spain, October 20 - 24, 2008). Q. Li, S. Spaccapietra, E. Yu, and A. Olivé, Eds. Lecture Notes In Computer Science, vol. 5231. Springer-Verlag, Berlin, Heidelberg, 1-11, 2008.

[37] Pastor, O.; Hayes, F.; and Bear, S. OASIS: an object-oriented specification language. Fourth International Conference on Advanced Information Systems Engineering, CAiSE 1992. Lecture Notes in Compute Science 593. Berlin: Springer-Verlag, 348–363, 2002.

[38] Pastor, O.; Molina, J.C.; and Iborra, E. - Automated production of fully functional applications with OlivaNova model execution. ERCIM News, no. 57 (April), 62–64, 2004.

[39] Rouff, C., Vanderbilt, A., Truszkowski, W., Rash, J. and Hinchey, M. Verification of NASA Emergent Systems. First IFIP International Conference on Artificial Intelligence Applications and Innovations. August 23-26, 2004.

[40] Schank, R. Tell me a Story. Northwestern University Press, 1990.

[41] Turner, M. The Literary Mind. Oxford University Press, 1996.

[42] Viterbo, J.; Mazuel, L.; Charif, Y.; Endler, M.; Sabouret, N.; Breitman, K.; Segrouchni, A.; Briot, J. - Ambient Intelligence: Management of Distributed and Heterogeneous Context Knowledge. In: Waltenegus Dargie. (Org.). Context-Aware Self managing Systems. Londres: CRC Press v. 1, p. 1-25, 2008.