A Systematic Relationship Analysis for Modeling Information Domains

Joonhee Yoo, Rutgers University
joonyoo@pegasus.rutgers.edu

and

Michael Bieber, New Jersey Institute of Technology
bieber@njit.edu; http://www-ec.njit.edu/~bieber

Abstract

Many conceptual modeling and system design methodologies provide tools to help system designers to model the real world. No guidelines exist, however, for determining the relationships within conceptual domains or implementations. RNA (Relationship Navigation Analysis), based on a generic relationship taxonomy, provides a systematic way of identifying useful relationships in application domains. Developers can then implement each relationship as a link. Viewing an application domain from the relationship management point of view and modeling from a philosophy of maximum access provides a unique vantage point for application design. We present RNA and its generic relationship taxonomy, describing their use for system analysis.

1. Motivation

When reengineering a legacy system for the World Wide Web or developing a new Web application, how does a systems developer determine what to link? A vital aspect of hypermedia system design is identifying relationships and implementing them as links [Fielding et al., 1998]. Yet, many relationships in applications—including analytic applications—are poorly identified or ignored in current hypermedia design methodologies [Isakowitz et al., 1995, Koufaris, 1998, Lange, 1994, Schwabe et al., 1996]. Furthermore, many Web applications do not take advantage of the major hypermedia features of the Web—linking, structural and navigational features. Few designers explicitly think about their applications' interrelationships and whether users should access and navigate them directly. This occurs for several reasons [Bieber and Vitali, 1997]. In part, existing applications demonstrate a rich link structure that could serve as examples for system developers. In part, few tools exist that help system developers to think of an application in terms of its relationships [Bieber, 1998a, Bieber, 1998b]. RNA (Relationship Navigation Analysis) was developed to solve these problems.

RNA can be used as part of a systems analysis, either to thoroughly describe an existing system (or information domain) in terms of its relationships, or to understand a system being designed. RNA provides systems analysts with a systematic technique for determining the relationship structure of an application, helping them to discover all potentially useful relationships in application domains. These later may be implemented as links. RNA also helps determine appropriate navigational structures on top of these links. RNA enhances system developers' understanding of application domains by broadening and deepening their conceptual model of the domain. Developers can then enhance their implementations by including additional links, metainformation and navigation.

In §2 we introduce the philosophy of maximum access and the hypermedia philosophy of design. §3 gives an overview of RNA's steps. §4 focuses on the third step: relationship analysis. §5 introduces RNA's generic relationship taxonomy. §6 describes a deeper layer of the taxonomy: the domain independent categories. §7 presents an example case study. §8 presents some future directions. We close in §9 with a review of the contributions of this research.

2. Hypermedia and Design

Hypermedia can be thought of as the discipline of relationship management [Isakowitz et al., 1995]. It considers a system in terms of the relationships among its elements and processes, focusing on how users gain access to them. This view of relationship management follows two philosophies: the "philosophy of maximum access" and a hypermedia philosophy of design. The philosophy of maximum access grants users full freedom to access and explore at will, helping them better understand a domain as a whole and build confidence in application results. Under this philosophy, any element of interest to a user should be a candidate for linking. Under the hypermedia philosophy of design, hypermedia analysis should play a part in the design of every application with user interaction. Also, hypermedia access should supplement many application's feature sets [Bieber, 1998b].

RNA provides a systematic approach to realizing a philosophy of maximum access within computer applications, supporting a hypermedia philosophy of design. RNA has the potential to establish new standards for designers in the application development process and for users' interaction with applications. Designers should do a relationship analysis in order to understand the relationship structure of their applications. Users should be able to point to any object of interest and find out whatever they want about it [Bieber, 1998b].

RNA employs hypermedia as a modeling tool not just an information model.

3. Overview of RNA Steps

A Relationship-Navigation Analysis can be conducted both for describing a system or domain (steps a - d) or for analyzing a system being designed (steps a - e). All steps can be performed in an incremental and iterative fashion.

A Relationship-Navigation Analysis comprises the following steps:

a. Stakeholder Analysis
b. Element Analysis
c. Relationship Analysis
d. Navigation Analysis

e. Implementation Analysis

3.a. Stakeholder Analysis

Stakeholders—anyone using some aspect of the application—are identified using an initial list of standard user types. An initial list could include software designers, systems analysts, technical authors, user representatives, training and user support staff, business/market analysts, and project managers [Macaulay, 1993]. Additional stakeholders are identified through the iterative process of goal, task, and domain analysis [Prieto-Diaz, 1990]. Also, stakeholders can be identified through the scenarios in the process of developing the requirements for the systems being developed.

3.b. Element Analysis

This step identifies what each stakeholder might want to find out more about. There would be different elements of interest for each class of stakeholders. For new systems, stakeholders' interests could be identified using mechanisms such as scenarios [Gotel and Finkelstein, 1995]. For existing systems, stakeholders' interests could be identified from the system's display screens.

Potential elements include:

- domain objects for which a definition, attribute value or other metainformation could be available, including events, products, executable programs, agents, commands, and parameters; [Kaindl, 1993]

- components (model, data, comments, and descriptions);

- aggregates of individual components (short unstructured lists, linear structures, array or tables, hierarchies, trees, paths, tours, webs, networks, and navigational aids (indexes, maps, table of contents, fisheye views)); [Garzotto et al., 1993, Parunak, 1990b]

- properties of entities such as attributes, component formats and structures, and network topology;

- spatial arrangements of icons in spatial hypermedia model (indicating an implicit relationship among them); [Nuernberg et al., 1998]

- rhetorical composites (specific constellations of nodes and links that form logical units for manipulation and navigation); [Parunak, 1990a]

- results of any operation (an explanation, calculation, and error messages).

3.c. Relationship Analysis

This step identifies relationships for each element of interest. In this step, each generic relationship prompts a series of questions which designers can ask themselves about each element of interest for each class of stakeholders. Table 2 in §5 shows RNA's generic relationships. Table 1 in §4 lists sample questions for each generic relationship.

Relationship analysis can be done at the three levels of detail shown in Figure 1. At the topmost general level, analysts use the broad generic relationship categories to examine each item of interest. Generic relationships are described using theoretical labels that capture the breadth of each generic category. The generic taxonomy includes several sub-levels, as shown in Table 2. While analysts can work at any of these generic sub-levels, we believe the bottom leaves will prove the most useful.

When analysts want to examine part of their domain more deeply for a particular generic relationship type, they can then use a more detailed set of categories at the domain-independent level. Domain-independent categories use more concrete, specific terms. §6 gives several examples.

Whereas the set of generic relationships is meant to be complete, over time our collection of domain-independent categories might approach only relative completeness. Researchers always can come up with new ways to subcategorize a generic category.

When analysts apply a domain-specific set of terms that maps to one of the domain-independent categories, they then are operating at the third level of relationship analysis.

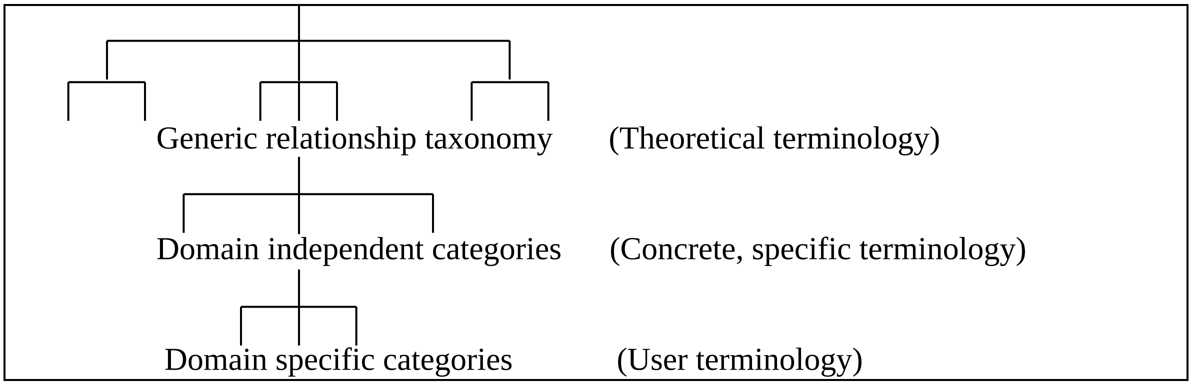

Figure 1: Generic, Domain Independent and Domain Specific Relationship Types

3.d. Navigation Analysis

This step identifies possible navigational structures for each stakeholder. In this step analysts think of each element of interest in terms of how the stakeholder might usefully access it. We can probe for additional relationships by asking navigational questions akin to the relationship questions in step c. For example, we could ask: "If this element were in an index/guided tour/trail for a particular stakeholder task, what else would be on that navigation structure?"

Note that being a brainstorming procedure, this step is different from a standard navigational analysis. It could feed into a navigational design in methodologies such as RMM [Isakowitz et al., 1995] and OOHDM [Schwabe et al., 1996]. They could use RNA's navigational analysis results to design appropriate navigational structures for sets of elements.

3.e. Implementation Analysis

In this step, analysts and designers do an informal cost/benefit analysis to decide which relationships would be useful and feasible enough to include in the application. Analysts should feel cognitively unbounded during the first four stages of the analysis, putting off practical constraints to this last stage.

Designers could decide to implement only those relationships which serve a recognized stakeholder or task (if deemed cost effective). During the RNA, however, other possible relationships are sure to have been thought of. In order to accommodate unexpected uses of an information system, designers might consider including additional relationships that appear to serve no current stakeholder task.

For the remainder of this chapter, we concentrate on step (c), the relationship analysis.

4. Conducting a Relationship Analysis

Within a generic information system, items of interest can be objects, people, commands, or even pieces of metainformation (if a stakeholder would want to focus on it). For each item of interest identified in stage 3b, an analyst employs relationship analysis as a knowledge elicitation or brainstorming tool, using it as a framework or set of categories with which to examine the application and its information environment.

Table 1 gives a series of possible questions which an analyst could ask himself or herself about each element of interest. Alternatively a facilitator could prompt an analyst or domain expert using these questions. We derived each set of questions from the domain independent categories described in §6. For the purposes of this chapter, the questions in Table 1 are rather condensed and highly generic. Obviously they should be tailored to each element of interest. For example, the descriptive relationship prompts designers to ask whether an item of interest has a definition, explanation, set of instructions or illustrations available within or external to the system. (These are all domain independent categories for the generic relationship "descriptive.") The analyst or facilitator clearly should ask each of the questions individually, and in a way that makes the most sense.

Generalization/ Specialization	Is there a broader term for this item of interest? Is there a narrower term for this item of interest?
Characteristic	What attributes and parameters does this item of interest have?
Descriptive	Does an item of interest have a description, definition, explanation, or a set of instructions or illustrations available within or external to the system?
Occurrence	Where else does this item of interest appear in the application domain? What are all uses of this item of interest?
Configuration/ Aggregation	Which components consist of this item? What materials are used to make this item? What is it a part of? What phases are in this whole activity?
Membership/ Grouping	Is this item a segment of the whole item? Is this item a member of a collection? Are these items dependent on each other in a group?
Classification/ Instantiation	Is this item of interest an example of a certain class? If a class, which instances exist for this element's class?
Equivalence	What is this item of interest equal or equivalent to in this domain?
Similar/Dissimilar	Which other items are similar to this item of interest? Which others are opposite to it? What serves the same purposes as this item of interest?
Ordering	What prerequisites or preconditions exist for this item? What logically follows this item for a given user’s purpose?
Activity	What are this item’s inputs and outputs? What resources and mechanisms are required to execute this item?
Influence	What items (e.g., people) cause this item to be created, changed, or deleted? What items have control over this item?
Intentional	Which goals, issues, arguments involve this item of interest? What are the positions and statements on it? What are the comments and opinions on this item? What is the rationale for this decision?
Socio-organizational	What kinds of alliances are formed associated with this item of interest? Who is committed to it in the organizational structure? Who communicates with it or about it, under what authority and in which role?
Temporal	Does this item of interest occur before other items? Does this item occur while other items occur?
Spatial	Which items is this item of interest close to? Is this item of interest nearer to destination than other items? Does this item overlap with other items?

Table 1. Sample questions for systems analysis for RNA's generic relationships

5. Generic relationships

Table 2 presents our generic relationship taxonomy.

Generic Relationship	Internal	Generalization/Specialization	
		Self	Characteristic Descriptive Occurrence
		Whole-part/Composition	Configuration/Aggregation Membership/Grouping
		Classification/Instantiation	
	External	Comparison	Equivalence Similar/Dissimilar
		Association/Dependency	Ordering Activity Influence Intentional Socio-organizational Temporal Spatial

Table 2. RNA's Generic relationships

5.a. The generic relationship taxonomy

Relationships can be categorized broadly as hierarchical vs. non-hierarchical [Neelameghan and Maitra, 1978]. We consider hierarchical as internal and non-hierarchical as external. We consider relationships that exist between an object itself and its characteristics or descriptions, and among different views, occurrences or transformations of the object to be internal.

Internal relationships can be broken down into self, generalization/specialization, whole-part/composition, and classification/instantiation relationships. This generalization/specialization, whole-part, classification/instantiation and association relationship classification agrees with the relationship classification in object-oriented analysis [Martin and Odell, 1995]. External relationships can be broken down into association/dependency and comparison relationships.

RNA's generic relationship taxonomy contains the following relationships:

- *Generalization relationship* – connects an item of interest to the items whose concepts include its concept in a taxonomy.

- *Characteristic relationship* – connects an item of interest to its attributes, parameters, metadata and other background information.
- *Descriptive relationship* – connects an item of interest to definitions, illustrations, explanations, and other descriptive information.
- *Occurrence relationship* – connects multiple instances/views/uses/transformations of the same object in different parts of a system.
- *Configuration/Aggregation relationship* – connects a part to other parts or a whole functionally or structurally.
- *Membership/Grouping relationship* – connects a member of a collection to other members or a whole collection.
- *Classification relationship* – connects an item of interest to its instance or class.
- *Equivalence relationship* – connects instances of the exact same object to a given item (i.e., same copies of a book or exact match in information retrieval).
- *Similar/Dissimilar relationship* – connects all items that share some positive or negative degree of similarity.
- *Ordering relationship* – puts items in some kind of sequence.
- *Activity relationship* – deals with relationships that exist among elements that are involved in some kind of activity (e.g., among an input, tools, and an output).
- *Influence relationship* – connects an item of interest to the item over which it has some kind of influence (i.e., causal or control relationship)
- *Intentional relationship* – connects an item of interest to the goals, arguments, issues, decisions, opinions, and comments associated with the item.
- *Socio-organizational relationship* – connects an item of interest to the position, authority, alliance, role, and communication associated with the item in a social setting or organizational structure.
- *Temporal relationship* – connects an item of interest to temporally related items.
- *Spatial relationship* – connects an item of interest to related items in spatial dimensions.

5.b.Global Relationship Attributes

As with the generic relationships, analysts can use global relationship attributes both to describe an existing system or analyze a new system to be designed. Global relationship attributes indicate the common properties that could be applied to all generic relationships. We have found the following global relationship attributes useful to think about:

- *Structural relationship* – any (generic, domain independent or domain dependent) relationship that connects related objects based on the application's internal structure. A structural relationship exists when another relationship is built into an application, either directly or indirectly. For example, in a relational database, each tuple is structurally related to the table in which it resides.

- *Implemented/Operation relationship* – any relationship representing menu options, command line operations, and other executable commands. This relationship connects an object to the result of operating upon it. (When a structural relationship is already implemented in an application, it also is an implemented/operation relationship.)
- *Schema relationship* – any relationship that explicitly exists in a system's design documents (e.g., a schema or functional decomposition diagram).
- *Statistical relationship* – any relationship giving access to any item occurring under similar conditions or otherwise statistically related to an item of interest. Related items found through data mining, for example are statistically related.
- *Process relationship* – any relationship representing processes and tasks, next step in a work flow, and subtasks in a project management system. A process relationship holds when the elements involved in relationships are connected through processes.
- *Coordinated relationship* – any relationship in which one element occurs automatically when other elements occur. [Evens, et al., 1980]
- *Coupling relationship* – joins two items if a modification to one of these items could, in general, requires that the other item should be checked for correctness and possibly modified, in order to preserve the consistency of the conceptual model is to be preserved. [Debenham, 1998]

Just as analysts and facilitators can use the questions prompted by the generic relationships in Table 1, they also can use global relationship attributes as a brainstorming tool to solicit additional relationships among domain elements. Table 3 shows sample questions for each.

Structural relationship	Which items are within this item of interest's internal structure? Is this item computed with any formula?
Schema relationship	Which items are associated with this item of interest in any application design document?
Implemented/Operation relationship	Is this item of interest a result of any computation? Which commands can be performed on this item, and with what consequences?
Statistical relationship	Does this item of interest tend to occur with other items with some probability?
Process relationship	Does this item represent one of steps intended to reach a goal?
Coordinated relationship	Does this item tend to occur together with other items?
Coupling relationship	Should any other items be checked for correctness if this item changes?

Table 3. Sample questions for global relationship attributes

5.c. Other relationship taxonomies

In developing our generic relationship taxonomy, we examined as many other taxonomies as we could. Each of their relationships fits somewhere in our generic taxonomy. Because many of the other taxonomies are domain-specific and quite detailed, by necessity the corresponding relationships in our generic taxonomy often are much broader categories. Our generic relationships could become a reference model for specific relationships, or could be expanded to accommodate specific relationships on a generic level.

6. Domain Independent Categories

Another goal of RNA is to provide a meaningful set of domain independent categories to further specify the nature of each generic relationship. Domain independent categories are developed using examples and dimension analyses of generic relationships.

This said, most domain independent categories often will not be used explicitly in a less-detailed relationship analysis. But analysts and developers could use the domain independent categories to delve deeply into a particularly fruitful relationship category. Thus the domain independent categories are most useful when conducting a very deep analysis, or for providing useful questions at the generic relationship analysis level (as with the questions in Table 1).

In this section we show a few of our domain independent categories. Note that whereas the generic relationship taxonomy is meant to be complete, by design the domain independent categories can never be complete.

a) *Domain Independent Categories in the Ordering Relationship*

Ordering	Series	Causal Derivation Generalization/Specialization Whole-part/Composition Topological Phase/Step/Stage	
	Rank	Preference Predominance Priority	
		Strictness	Mandatory Optional Default

		Importance Relevance Recency Alphabetical Numerical Similarity Temporal Distance
	Adjacency	Stack Queue

The ordering relationship has three domain independent subcategories (series, rank, and adjacency). Each subcategory has its own subcategories. Series covers ordering that are identified in a series or chain reactions of events. Rank deals with orderings that exist in some kind of scale. Strictness, one of rank's subcategory, has its own subcategories indicating the degree of strictness [Conradi and Westfechtel, 1998]. Adjacency covers orderings that are identified in some kind of structures, such as stack, queue, and the periodic table [Schubert et al., 1983]. These examples come from many domains, including bibliographic, information retrieval, temporal, geographic, hypermedia, statistical, and data structures.

b) Domain independent categories in ther Descriptive relationship

Descriptive	Explanation Illustration Definition Clarification Elaboration Evaluation Review Detail Summary Demonstration Instruction

The domain independent category for the descriptive relationship includes examples from description or explanation related activities associated with an item of interest.

c) Domain independent categories in the Configuration/Aggregation relationship

Configuration/Aggregation	Component-Integral/Assembly Material-Object Place-Area Feature-Activity

The configuration/aggregation relationship has four domain independent categories. All are structurally or functionally related among an element's parts or between its parts and the whole. The whole-part relationship has established sub-relationship taxonomies in many disciplines [Iris et. al., 1988, Motschnig-Pitrik, 1993, Odell, 1994, Winston et al., 1987]. The four categories we use are from the classification done in [Firesmith and Henderson-Sellers, 1998, Henderson-Sellers, 1997], and encompass the others.

7. An Example Case Study

We recently applied RNA to the domain of on-line bookstores. We illustrate here with some of the relationships RNA helped us find for the element "book." Some of the relationships we found are already implemented in bookstore Web sites, but many do not appear there. An RNA implementation analysis (step 3e) would show that many either are not cost effective to provide and others might give users access to competitors, which an e-commerce Web site often will avoid. Yet, some are useful. And several of the others that the RNA implementation analysis by a bookstore would reject, provide opportunities for third parties to sell or governmental services to provide to benefit the common good. In any event, we were amazed at the scope of relationships we found that do not appear on the Web, yet seem so obvious once we performed the RNA analysis.

Generalization/Specialization

Using the specialization relationship, we determined that a book is an abstraction of the objects novel and short story. Often customers have a preference for collections of short stories or for full novels.

Using the generalization relationship, we realized that books could serve several different roles. For example, books could be generalized into "reading materials," and that many other kinds of reading materials exist besides books that an on-line bookstore could provide. Books also are a kind of product, and that an on-line bookstore could consider other kinds of products such as videos. These roles vary based on the customer's intent (looking for something quick to read, looking for something for a long trip, looking for a gift, looking for something to amuse me this evening), and an on-line bookstore could expand to serve several of these intents.

Characteristic

Using the generic characteristic relationship led us to the following characteristics of books, in which different customers might be interested:

- condition (new/used/damaged)
- relevance (How long will this book be relevant? For example, a road map or set of statistics might be valid for a month, a year or a decade.)
- date published, edition, whether this a reprint of a recent or an old work
- date written (Did it take long to publish? Is this a reprinting of a lost or classic work?)
- owner (Who owns the copyright on this work?)
- contributors (authors, illustrators, editors, people interviewed during its authoring)
- level of reading skills required
- intent (reference, history, how to, self help, tutorial, etc.)
- typeface (fonts used)
- type of print (large print, Braille, audio)
- awards received
- ratings (from different consumer groups)

- dimensions, weight (dictating the cost of shipping)
- price (wholesale, retail, discount, bulk, educational discount)

Occurrence
Both customers and systems analysts will be interested in various occurrence relationships for books:

- Where is this book listed? (best seller lists)
- Where has this book been reviewed or discussed?
- Are there translations available?
- Are there newer/older/early/draft versions available, perhaps under a different name?
- Does this book have prequels or sequels?
- Who (else) sells this book?

The occurrence relationship also leads to warnings an on-line bookstore could provide:

- "You already have a copy of this book in your shopping basket. Are you sure you want another?"
- "You purchased this book last week, but it has not been delivered yet."
- "You purchased this book last December."

Configuration/Aggregation
Using the domain independent categories for the generic <u>configuration/aggregation</u> relationship, we determined that a book is related to the following objects that it contains:

- its chapters (Is the table of contents available? Can the customer read the first chapter?)
- its index (giving an indication of the book's level of detail and expertise)
- its forward (which might entice a customer)
- its introduction (giving an indication of the book's level of detail and expertise)
- its illustrations (customers may be enticed by the illustrations in a children's book or figures in a technical book, for example)

Using the <u>configuration/aggregation</u> relationship, we determined that a book may also be a part of a series. The customer may wish to see other books in the series.

Process
The global <u>process</u> relationship attribute led us to ask how the following processes work. Using good interface design, an educational description of each could be accessible from any book object:

- how a book is written
- how a book is published
- how a book is manufactured

Activity
The generic <u>activity</u> relationship leads us to ask who and what uses books, and how:

- which kinds of people read a certain book (Which types of customers might want to buy this book?)
- people give books as gifts (Is the bookstore's Web site set up to facilitate people looking for gifts?)
- book groups (Is the bookstore doing anything to support book groups?)

Using the generic activity relationship we also determined which objects are inputs to a book:

- paper (Is the paper good quality? Is it printed on recycled paper?)
- binding (Was the book manufactured to last?)
- cover (Is it hardcover or softcover?)

The generic activity relationship also prompts us to ask which activities a book results from:

- result of research
- result of a journey
- result of a crisis in the author's life

Influence
The generic influence relationship leads us to ask which people, events, philosophies, and other books might have influenced the author or the subject matter of a book. Customers fascinated by a book (or author) might want to learn more about these influences.

8. Future research

Future research on RNA ranges from improving the analysis to extending the approach.

8.a. Scaling RNA

Not all relationship types will apply to every application, and developers might not have time for a full RNA analysis. Thus we plan to scale RNA to serve different purposes.

As part of scaling, we need to determine which relationship types are most likely to be useful on which general kinds of elements of interest, for which general kinds of domains, and for which general kinds of stakeholders. It might be desirable to have multiple domain independent categories for each generic relationship to meet the needs of different situations.

8.b. Integrating RNA with existing tools

RNA could be a part of any requirements analysis in system development methodologies. We shall investigate seamlessly integrating RNA into established system design methodologies [Kruchten, 1995], including hypermedia design methodologies [Christodoulou et al., 1998].

8.c. Design patterns

Each generic relationship in RNA could be represented as specific design or analysis patterns [Gamma et al., 1995, Coad, 1992, Nanard et al., 1998]. This would provide us a clear and well-defined technique for expressing the relationships. Furthermore, the standard pattern type representation could lead to a relationship language that would facilitate automated generation of links within application designs.

8.d. Potential for RNA of any complex system

Much of human knowledge builds upon comparing objects. The generic relationships in RNA certainly apply to domains outside of the information domain to some degree. In a sense, all the relationships in the real world are candidates that could be covered by the generic relationships in RNA. Thus we shall investigate whether RNA could serve as an analysis technique to determine the relationship structure of any complex system, not just computer applications.

9. Discussion & Contributions

RNA's generic relationship taxonomy is theoretically robust. It is based on existing theories, ontologies, information modeling methodologies and incorporates established taxonomies logically. It covers the various aspects of information modeling including subject world, system world, usage world, and development world [Mylopoulos et al., 1990]. Also, we believe it is complete in the sense that it is comprehensive enough to cover any existing taxonomy (or at least all we have examined thus far). Generic relationship analysis is unique in that it deals with generic domains.

RNA's generic relationship taxonomy provides a kind of relationship checklist for any application domain. RNA could be combined with conceptual modeling tools to provide a rich conceptual modeling environment where different kinds and levels of abstraction as well as diverse relationships could be supported according to stakeholders' interests RNA provides additional conceptual modeling concepts beyond usual abstraction mechanisms (generalization/specialization, whole-part, classification) enabling rich, relationship-based semantics to be incorporated into information modeling support mechanisms. RNA contributes to hypermedia design methodologies by providing a systematic way to find a comprehensive set of linking and navigational opportunities. In addition, RNA contributes to metadata research by identifying useful generic relationships that could be considered a rich form of metadata.

Yet, in the end we hope that our most enduring contribution is successfully convincing developers of Web applications (both new and transported from other computer environments) to take full advantage of linking in their applications.

Acknowledgements

We gratefully acknowledge the assistance of Fernando Lyardet and Gus Rossi (co-developer of OOHDM) of the University of La Plata, Argentina, in performing the RNA analysis of the bookstore domain. We also appreciate funding support for this research by the NASA JOVE faculty fellowship program, by the New Jersey Center for Multimedia Research, by the National Center for Transportation and Industrial Productivity at the New Jersey Institute of Technology (NJIT), by the New Jersey Department of Transportation, and by the New Jersey Commission of Science and Technology, and by Rutgers University.

References

Bieber, M. (1998a) Hypertext and Web engineering. Proceedings of Hypertext’98. pp. 277-278.

Bieber, M. (1998b) Supplementing applications with hypermedia. Under submission to ACM transactions on Information Systems, 1998.

Bieber, M., Vitali, F., Ashman, H., Balasubramanian V., and Oinas-Kukkonen, H. (1997) Fourth generation hypermedia: some missing links for the World Wide Web, International Journal of Human Computer Studies 47, 1997, pp 31-65.

Christodoulou, S., Styliaras, G. and Papatheodourou, T. (1998), Evaluation of hypermedia application development and management systems, ACM Hypertext'98 Conference, Pittsburgh, pp 1-10

Coad, P. (1992) Object-Oriented Patterns. Communications of the ACM, 35(9), pp 152-159

Conradi, R. and Westfechtel, B. (1998) Version Models for Software Configuration Management. ACM Computing Surveys, 30 (2), June 1998, pp 232-282.

Debenham J. (1998) Knowledge Engineering: Unifying Knowledge Base and Database Design.

Dominguez, E., Zapata, M.A., and Rubio, J. (1996) A Conceptual Approach to Meta-Modeling. Proceedings of the 8th International Conference on Advanced Information Aystems Engineering, CaiSE’96, pp 319-332.

Evens, M., Litowitz, B., Markowitz, J., Smith, R., and Werner, O. (1980) Lexical-Semantic Relations: A Comparative Survey.

Fielding, R., Whitehead JR., E. J., Anderson, K., Bolcer, G., Oreizy, P., and Taylor, R. (1998) Web-Based Development of Complex Information Products. Communications of the ACM, 41(8), 84-92.

Firesmith, D. and Henderson-Sellers, B. (1998a) Clarifying Specialized Forms of Association in UML and OML Journal of Object-oriented programming, May 1998, pp 47-50.

Firesmith, D. and Henderson-Sellers, B. (1998b) Upgrading OML to Version 1.1: Referential Relationships. Journal of Object-oriented Programming, June 1998, pp 48-57.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995) Design Patterns: Elements of Reusable Object-oriented Software. Addison-Wesley, Reading, MA, 1995.

Garzotto, F., Mainetti, L. and Paolini, P. (1993) HDM2:Extending the E-R approach to Hypermedia Application Design. Proceedings of the Twelfth International Conference on E-R approach ER'93, pp 179-189.

Garzotto, F., Paolini, P. and Schwabe, D. (1993) HDM – A Model for the Design of Hypertext applications. ACM Transactions on Information Systems 11, 1(Jan. 1993), pp 1-26.

Gotel, O. and Finkelstein, A. (1995) Contribution Structures. Proceedings of IEEE International Symposium on Requirement Engineering, pp 100-107.

Henderson-Sellers, B. (1997) OPEN Relationships-Compositions and Containments. Journal of Object-Oriented Programming, November/December 1997, pp 51-72.

Iris, M., Litowitz, B. and Evens, M. (1988) Problems of the part-whole relation. Relational Models of the Lexicon: Representing knowledge in semantic networks. Ed. Evens, M. Chapter 12, pp 261-287.

Isakowitz, T., Stohr, E. and Balasubramanian, P. (1995) RMM: A methodology for structuring hypermedia design. Communications of the ACM 38(8), Aug. 1995, pp 34-44.

Kaindl, H. (1993) The missing link in requirements engineering. ACM SIGSOFT Software Engineering Notes vol. 18 no. 2, Apr. 1993, pp 30-39.

Koufaris, M. (1998) Structured design of WWW and Intranet applications. The Seventh International WWW Conference Tutorial.

Kruchten, P. (1995) The 4+1 View Model of Architecture. IEEE Software, November 1995, pp 42-50.

Lange, D. (1994) An Object-Oriented Design Method for Hypermedia Information Systems. Proceedings of the Twenty-seventh Annual Hawaii International Conference on System Sciences, 1994, pp 366-375.

Leveson, N. (1998) Intent Specifications: An approach to building human-centered specifications. Proceedings of IEEE Conference on Requirement Engineering, pp 204-213.

Macaulay, L. (1993) Requirements Capture as a Cooperative Activity. IEEE International symposium on Requirements Engineering, San Diego, California, IEEE Computer Society Press, pp 174-181.

Martin, J. and Odell, J. (1995) Object-Oriented Methods: A Foundation. Prentice Hall, Englewood Cliffs, New Jersey, 1995.

Motschnig-Pitrik, R. (1993) The Semantics of Parts Versus Aggragates in Data/Knowledge Modeling. Advanced Information Systems Engineering, Proceedings of the 5th International Conference CAISE’93, pp 352-373.

Mylopoulos, J., Borgida, A., Jarke, M., and Koubarakis (1990) Telos: Representing Knowledge About Information Systems. ACM Transactions on Information Systems, Vol. 8, No. 4, October 1990, pp 325-362.

Nanard, M, Nanard, J. and Kahn, P. (1998), "Pushing reuse in hypermedia design: Golden rules, design patterns and generic templates," Hypertext '98 Proceedings, ACM Press, Pittsburgh, pp 11-20.

Neelameghan, A. and Maitra, R. (1978) Non-hierarchical associative relationships among concepts: Identification and Typology. Part A of FID/CR Report no. 18, Bangalore: FID/CR Secretariat Document Research and Training Center.

Nuernberg, P., Wiil, U., and Leggett, J. (1998) Structuring Facilities in Digital Libraries. Proceedings of the Second European Conference on Digital Libraries (ECDL ’98).

Odell, J. (1994) Six different kinds of composition. Journal of object-oriented programming, January 1994, pp 10-15.

Parunak, H. Van Dyke (1990a) Toward a Reference Model for Hypermedia. Proceedings of the Hypertext Standardization Workshop: Jan., 1990, National Institute of Standards and Technology, pp 197-211.

Parunak, H. Van Dyke (1990b) Reference and Data Model Group(RDMG):Work Plan Status. Proceedings of the Hypertext Standardization Workshop: Jan., 1990, National Institute of Standards and Technology, pp 9-13.

Prieto-Diaz, R. (1990) Domain Analysis: An Introduction. ACM SIGSOFT, Software Engineering Notes, 15(2), Apr. 1990, pp 47-54.

Schubert, L., Papalaskaris, M., and Taugher, J. (1983) Determining Type, Part, Color, and Time Relationships. IEEE Computer October 1983 pp 53-60.

Schwabe, D., Rossi, G., and Barbosa, S. (1996) Systematic hypermedia application design with OOHDM. Proceedings of Hypertext ’96, Washington DC, pp 116-128.

Winston, M.E., Chaffin, R. and Herrmann, D. (1987) A taxonomy of part-whole relations. Cognitive Science 11, pp 417-444.