Design Considerations for Information Systems to Support Critical Infrastructure Management

Madhavi M. Chakrabarty
New Jersey Institute of Technology, USA
mc52@njit.edu

David Mendonça
New Jersey Institute of Technology, USA
mendonca@njit.edu

ABSTRACT

This paper develops a set of design considerations for information systems to support the management of interdependent critical infrastructure systems. Constraints on how these systems are managed are oriented along technical, political and organizational dimensions, though objectives along these dimensions may conflict and thus be difficult to satisfy. This paper harnesses methodologies from software engineering and cognitive science in order to specify opportunities for using information systems to support human-centered management of critical infrastructure systems. The particular focus of this work is on developing information systems to support visualization and visual problem solving. Progress to date is discussed in terms of an ongoing research project which uses as a test-bed data associated with lower Manhattan (New York, USA).

Keywords

Critical infrastructure systems, software analysis and design, decision support systems

INTRODUCTION

Critical infrastructure systems are components of complex interdependent systems that are vital for the economic well-being and security of a nation and its citizens with physical or logical connections amongst each other (Wallace et al., 2003). The management of such systems involves coordination efforts amongst multiple and diverse participants or stakeholders (Peerenboom et al., 2001) over space and over time. This diversity among the stake holders and their organizations leads to numerous policy issues and organizational issues which hinder communication and collaborative efforts between decision managers. Policy and organizational issues include differences in protocols, rules, procedures and other guidelines followed by an organization, at an individual or group level, for tasks like information access, information flow and decision making.

Given these complexities and interdependencies, management of interdependent critical infrastructure systems is likely to require a variety of tools. Visual tools can be defined as collection of symbols graphically linked by mental associations to create a pattern of information and a form of knowledge about an idea (Hartley, 1996). Such tools reduce the cognitive load associated with information presentation and processing, and have been used in various applications (Card, 1999, Ware, 2000). Yet improved methods are needed for constructing visual tools for the management of interdependent infrastructure systems (Chakrabarty and Mendonça, 2004). This paper draws upon visualization and modeling sciences and tries to integrate system development perspectives to help design and develop a visualization-based information system for managing interdependent critical infrastructure systems. The particular focus of this paper is on developing systems used in supporting decision makers in restoring critical infrastructures that have been disrupted following disasters.

BACKGROUND

Critical infrastructures such as electric power and transportation are crucial systems for a nation's security and economic well-being (U.S. General Accounting Office, 2001). They are now viewed as interconnected and interdependent systems of systems (Peerenboom et al., 2001), in which an incident in one system is accompanied by or induces a disruption in one or more other systems. Various types of interdependency may occur (Wallace et al., 2003), and, as shown during the aftermath of the 2001 World Trade Center attack (Mendonça et al., 2004), a wide variety of infrastructures may be involved in these interdependencies. As an example of "input" interdependence, consider a telecommunications company that is responsible for a switching station. The switching station is used to route calls through the network. Power from an electric utility's transformer is required to operate the switching station, thus creating an input interdependency from power to telecommunications. An incident involving loss of power in the power system would therefore lead to a disruption in the telecommunications system.

Critical infrastructure systems go beyond physical systems. Apart from the physical infrastructure, there is also the information infrastructure (Luiijf and Klaver, 2004). The extent and usage of these systems have grown by leaps and bounds over the last decade. Research work to identify and define guidelines to select critical interdependent systems now includes information systems as well (Luiijf and Klaver, 2004).

Because critical infrastructures are interconnected, identification and management of disruptions is likely to be accomplished by two or more organizations, implying a coordinated decision-making amongst various public and private organizations and the need for interfaces between them (Chakrabarty and Mendonça, 2004). The development of infrastructure management systems will therefore involve multiple and diverse technologies, policies and organizations (Luiijf and Klaver, 2004), increasing problem complexity. A key management problem is therefore the integration of heterogeneous data sources, reasoning capabilities and resources in order to restore the services provided by critical infrastructures. Such problem solving is likely to require considerable human involvement and may not be wholly automated. A key design question, then, is how to accomplish integration by supporting human decision makers with information technology.

Human reasoning capability is highly sensitive to problem representation. Indeed, visually or lexically different representations of the same problem can dramatically influence problem solving behavior (Simon and Hayes, 1976). Human problem solving may, for certain problems, be improved through visualization, in which elements of the problem are represented visually (e.g., with diagrams or charts)(Larkin and Simon, 1987). It is therefore worth exploring how problem solving related to the management of critical infrastructures can be supported through visual representations.

Recent advances in computing and Internet have further increased the reach and richness of visualizations (Huang and Worboys, 2001). The introduction and advance of multimedia in computing has further led to better ways of rendering features on the computer, thereby influencing the user's cognition and action. Display techniques like movement, highlighting and salient icons have increased the richness of the media, message and modality (Sutcliffe, 1999), thus improving the presentation and cognitive richness of visualizations. Design of the user interface for the visual systems should be able to benefit from these improvements in hardware and software developments. The interface that the decision makers use should be able to match the media to the message, manage users' attention, navigation and control (Sutcliffe, 2003).

Visual models of infrastructure systems represent physical systems and their functionality in graphical terms, thus providing an interface for decision makers to interact with the underlying system as if they are acting directly on the system. They may be contrasted with lexical models, which describe infrastructures using the vocabulary and semantics of human language, and with mathematical models, which describe infrastructures using mathematical notation. An advantage of visual models over mathematical or lexical models is their increased interpretability. They may also require a lower cognitive load to use. Two disadvantages associated with visual models are (i) the lack of formal methods for their construction and validation and (ii) the difficulties involved in expressing them lexically or mathematically. The remainder of this section focuses on addressing this first disadvantage.

Information systems analysis and development methodologies provide numerous frameworks to improve user productivity and quality as well as system reliability and validity. Reliability is defined as the tendency towards consistency found in repeated measurements or observations of the same phenomenon (Carmines and Zeller, 1979). Validity is the extent to which any measuring instrument measures what it is intended to measure (Carmines and Zeller, 1979). A reliable and valid measure increases trust in the system and subsequently its usability. Visualization may be said to function as a measure, since it is intended to capture salient elements of the phenomenon being visualized (e.g., an infrastructure system). Reliability and validity are denoted as non-functional attributes of a system, since they are in effect measures of model quality and not model functionality. System development methodologies now include steps for specifying attributes of such non-functional attributes, as well as functional attributes such as system behavior for a given input, system processing for that input and expected results (Sommerville, 2001). Including exception handling in the design phase rather the coding phase is one such example (Ryder and Soffa, 2003).

Various technical, organizational and political factors that impact system design or usage must be identified and analyzed. Technical issues include data representation, data storage, data sharing, data validity and data security. Political issues include policy-level constraints and objectives, such as how to balance recovery costs against service restoration goals. Organizational issues include data sharing protocols and negotiation over shared resources (Sommerville, 2001), as well as behavioral issues such as how managers search for and use information (Vandenbosch and Huff, 1997, Sommerville, 2001). Information seeking behavior is a key concern. Managers look for information either by searching or scanning depending on various factors like nature of problem, time at hand, amount of information available and kind of expertise at hand (Vandenbosch and Huff, 1997). Any system intended to help the managers should be able to provide them with an interface to search, scan and transparently integrate between the two without additional load to him/her. These different usage patterns of managers under different circumstances also become additional issues that need to be addressed for designing considerations of the system specification. The storyboard approach to interface design (Sutcliffe, 1999, Nielson, 1995) provides an example of how an informal means of modeling for design can be implemented once the requirements are formalized.

RESEARCH

This section reviews progress to date in harnessing methodologies from software engineering and cognitive science in order to specify opportunities for using information systems to support human-centered management of critical infrastructure systems. The particular focus of this work is on developing information systems to support visualization and visual problem solving.

In software engineering, system development consists of various stages such as requirement elicitation, requirement formalization, system design, validating and testing (Sommerville, 2001). This paper lays stress on the first of these stages. This analysis is expected to lead to some effectiveness and expressiveness criteria which can be used in evaluating the specifications and providing groundwork for the development of visualizations. A guiding principle in the design of the visualizations is that the visualizations should be adequate, inexpensive and adaptable (Spoerri, 1999, North and Shneiderman, 2000). Visual components in a visual model should also be capable of abstracting information for easy comprehension and avoiding information overload. These characteristics of a visual model help to standardize and stabilize the model over time. The target users of the system represent multiple perspectives (e.g., of the managers of the constituent systems, as well as of those responsible for overseeing the connections amongst the various systems).

Requirements Specification

Requirements engineering (RE) is concerned with the identification of the goals to be achieved by the envisioned system. The processes involved in RE include domain analysis, elicitation, specification, assessment, negotiation, documentation, and evolution (Lamsweerde, 2000). Requirements engineering must address the contextual goals of why a software is needed, the functionalities the software has to accomplish to achieve those goals, and the constraints restricting how the software accomplishing those functions is to be designed and implemented and the social environment where the system is deployed (Lamsweerde, 2000, Goguen, 1996). The functional requirements of the target system constitute the features desirable by the end users of the system. These features include the user interface, various functionalities and system properties, along with the way they are implemented. Interface designs include people, tasks, organizations as well as the context of usage. Requirements also result in formalizing non-functional specifications including reliability, availability, accuracy and validity (Lamsweerde, 2000). Constraints may be technical, organizational or political. The scope of the model must also be determined. As shown by numerous power blackouts in the U.S. (U. S.-Canada Power System Outage Task Force, 2004), for example, even tenuous connections between systems can provide pathways for failure.

Software Development Methodology

Methodologies in software development provide guidelines for the development of various types of information systems. Systems design perspective has moved from a generic approach to more specific, task oriented, domain related and focused methodology based on specific and unique problems and scenarios. Guidelines are developed to provide guidance and reference for designers. But these guidelines tend to be stated at either a highly abstracted level making them vague and generic and difficult to interpret or at the level of interface widgets making it difficult to design interaction strategies for different users. Therefore, most available guidelines of system engineering for system development provide weak support and therefore is not sufficient to support developers faced with specific interface design problems targeted for specific user groups (Henninger et al., 1995). According to a research in Europe, instances are found where information security issues neglect the involvement, perspective and actions of policy makers as well as the critical information infrastructure stakeholders (Luiijf and Klaver, 2004). Technology-centric guidelines focus more on the platform presentation and widgets. On the other hand, generic guidelines that try to make the guidelines universal lose the very motivation in their attempt to generalize. A model, the software maturity model, presents a description of how systems evolve from the research phase through to operational deployment (Redwine and Riddle, 1985). Therefore, rather than assuming either a techno-centric or a generic guideline, a tailored approach suitable for the particular individual problem needs to be followed (Nielson, 1992, Hartson and Hix, 1989). A case-based approach using usability principles illustrated by examples (Henninger et al., 1995) is pursued in this paper. The use of experience-based usability guidelines (Henninger et al., 1995) enables the inclusion of organizational memory into the information system as the form of a repository. This kind of a repository helps in determining a suitable guideline to get what information is applicable and when by linking guidelines as well as examples stored in the organizational memory. It also supports reuse of domain-specific components or the specifications of requirements on a case-by-case basis.

The focus of development of the visualization for interdependent critical infrastructure systems is directed towards aiding decision making activities in managing these systems. This includes strategies for managing emergencies and the preemptive policies of mitigation and preparedness by reducing vulnerability as well as post-event response activities like containment and restoration. Given that interdependent critical infrastructure systems may operate under environments that are subject to disaster, the design of such systems should include adequate backup and redundancy measures. Though, the design of the actual systems may have modes that lead to failures, modeling the interconnections among the systems may help the managers of the systems develop innovative ways to determine alternate feasible solutions.

Method

The software maturity model (Redwine and Riddle, 1985) mentioned previously is particularly useful in discovering processes, principles and techniques in the development and implementation of a system. This model describes the steps leading to transition from a research concept to widespread use of tools. Since the current research deals with a wide range of systems managed by a diverse range of people, it is desirable to adapt this guideline and use the technique to ensure the usage of the system by all the stakeholders.

The following are the steps identified by the model.

- Step 1: Basic Research. Investigate ideas, concepts to be included in the system and identify critical ones
- Step 2: Concept Formulation. Identify key problems, circulate ideas, identify issues and start to develop solutions.
- Step 3: Development and Extension. A clear solution emerges, Form prototypes, begin preliminary usage of the system, and enable a clear solution to emerge.
- Step 4: Internal Enhancement and Exploration. Extend the approach to other domains and stabilize system. Develop training and user guides.
- Step 5: External Enhancement and Exploration. Extend the approach to a broader group and move it outside the development group.
- Step 6: Popularization. Investigate evidence for and against the applicability and reach of the software; implement production and commercialization.

It is evident that usability principles cannot be separated from software engineering methods. The visual model acts as the interface of the system for the user and provides a way to access the system details and services. Therefore, the target system should not only be capable of attracting the users to use it but also help the decision makers accomplish their task. Combined with the case based approach mentioned earlier (Henninger et al., 1995), the software maturity model provides a good framework which can be adapted for developing visual components, and hence derive the model. This line of system development perspective helps integrate the motivation of building the tool while taking into consideration the various perspectives of the stakeholders leading to better use of the system.

RESULT

The phases (steps) derived from the software maturity model are here applied to an example scenario in order to develop a visual model for interdependent structures. The following example (adapted from (Wallace et al., 2003)) involves interdependent telecommunications and power infrastructures. Assume that a telecommunications company is responsible for a switching station, which is used to route calls through a network. Power from an electric utility's transformer is required to operate the switching station, creating an input interdependency from power to telecommunications. An electric transformer failure results in power loss at the telephone switching station, which in turn causes subsequent disruption of the telephone service. A Supervisory Control And Data Acquisition (SCADA) system is available to monitor the system status for a group of transformers and other power generating and transmitting units.

Applying the software maturity model, in Step 1 the basic problem has to be identified. This involves investigating the failure in the system (i.e., root cause and important effects), mitigating the impacts and expediting restoration efforts. The goal of this step is to develop the visual model for depicting the system's state. In Step 2, once the problem is identified, it has to be communicated among appropriate stakeholders (e.g., government and utility officials, and representatives of the public). Including stakeholders' inputs should reduce communication barriers and enrich collaboration.

Step 3 requires building an initial prototype of the system which depicts the system from possibly many perspectives. This helps to abstract out information that cannot be manipulated by a particular stake holder. For example, the electric power personnel need the details of disruption, the trigger (such as overheating of the transformer) that causes a power failure and the degrees of system disruption. Other stakeholders need to know the source of the problem (i.e., the power supply node). But the support personnel responsible for the supply node need to know the exact location or reason for the failure. Extension of the prototype stage includes internal extension of the model across the individual organizations (Step 4) and external extension over the interconnected systems (Step 5) to develop a complete visualization. Popularization (Step 6) leads to an increase in the reach of the tool. Test runs of the visualization with decision managers will enable increase the reach and hence the popularity and commercialization of the tool.

An information system's usability and success depend in part on the non-functional requirements of reliability and validity. Therefore, it is necessary to test these parameters with respect to the system created by following the design procedure. Empirically, reliability can be measured with various methods (Carmines and Zeller, 1979), but these are difficult to adapt to complex models. Validity may be addressed through focus groups (which can provide critiques on the representation of the infrastructure systems) and through system simulations (Peerenboom et al., 2001).

Capturing organizational memory with the information gathered in the previous steps helps to develop a knowledge base which may include guidelines for system development as well as examples to illustrate the implementation of the guideline in a particular scenario. This methodology incorporates understanding the practices of different organizations so that new practices and guidelines can be easily incorporated into the organizational framework over time. This methodology incorporates into it a primitive form of evolutionary improvement process to refine previously developed guidelines. The knowledge base can also be helpful when new guidelines emerge for specific instances in an organization. The knowledge base can in fact trigger the trend towards a defined standard.

DISCUSSION

Design of a complex system with multiple stakeholders may include consideration of a number of factors, including the following (Spoerri, 1999). The system should be *adaptable* to different perspectives of different users. It provides the details of only that component which the user has permissions to manipulate and it can provide an overview of the entire system for establishing the context. The system should *cohere* (i.e., it should enable integration of its components into another representation), which also implies ease of changing views while maintaining the problem, its scope and the context. It should *capture the dynamics* of the system, so that requirements are reflective of current needs. It should enable *unified specifications*: different organizations have different standards and policies to follow and hence an effort to integrate these specifications should be made.

Organizations may have disagreements and conflicts in designing interfaces that attempt to integrate their systems. Asking each stakeholder group to develop a list of interfaces to be opened up to other stake holders as well as the interfaces external stakeholders need to open up to them will enable them to reach a consensus in formalizing interface specifications. These inputs from the stakeholders will also help in reducing conflicts, thereby increasing collaboration efforts amongst them. Developing the visualization on the lines of software detailed above is expected to produce a system with features desirable to the decision makers (users).

System development, software engineering and usability design principles are no longer considered in isolation. The implication of the cumulative effect of these three perspectives of system development is particularly strong when the stakeholders belong to diverse organizations. Monitoring of interdependent critical infrastructure systems provides a similar platform where multiple decision makers interact with infrastructure systems that belong to them but are regulated and used by others. Representations—whether lexical, mathematical or visual—of such infrastructure systems require a high degree of reliability and validity. The approach set forth in this paper extends a software development methodology to the design of visualization for interdependent critical infrastructure system and reinforces it with knowledge based usability design principles.

CONCLUSIONS AND FUTURE WORK

Design consideration for any socio-technical system should have a human centered approach. In the present context of interdependent critical infrastructure systems, this design problem leads to the development of design guidelines to develop a visual model of such systems. Such a model should support decision makers and managers of interdependent critical infrastructure systems in reducing their cognitive load and identifying core elements of the problem.

This paper focuses on a software development method for developing visualizations of interdependent critical infrastructure systems. It presents requirements and design principles for a set of requirements which is volatile and dynamic and multi-faceted due to the large number and scope of diverse stake holders. It also addresses ways to reduce the conflicts arising due to inclusion of multiple perspectives of different stakeholders in different systems and their interconnections to set a pace for development of such systems.

Evaluation of the reliability and validity of the visual models produced using the methodology presented here is an important area for future research. Evaluating the validity of the theory and the usability of the model will ensure the soundness of the theory of the application of the design methodology to the area of interdependent systems. A prototype of the system needs to be developed before any empirical analysis can be done. A group of experts having experience in dealing with interdependent infrastructure systems can be contacted to set up a plausible scenario for the present work. The results of the empirical analysis will justify the validity of the method.

ACKNOWLEDGEMENT

This research is being supported by National Science Foundation Grant CMS-0301661.

REFERENCES

1. Card, S. K. (1999) Readings in Information Visualization: Using Vision to Think, Morgan Kauffman.
2. Carmines, E. G. and Zeller, R. S. (1979) *Reliability and Validity Assessment,* Sara Miller McCune, Sage Publications, Inc.

3. Chakrabarty, M. and Mendonça, D. (2004) In *IEEE International Conference on Systems, Man and Cybernetics*The Hague, The Netherlands.
4. Goguen, J. (1996) In *Proc. IEEE International Conference Requirements Engineering*IEEE CS Press, Los Alamitos, California.
5. Hartley, J. (1996) *Text Design,* Simon & Schuster Macmillan, New York.
6. Hartson, H. R. and Hix, D. (1989) Human-Computer Interface Development: Concepts and Systems for Its Management.*ACM Computing Surveys,* **21,** 5-92.
7. Henninger, S., Haynes, K. and Reith, M. W. (1995) A Framework for Developing Experience-Based Usability Guidelines.*Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods and Techniques***,** 43-53.
8. Huang, B. and Worboys, M. F. (2001) *Dynamic Modeling And Visualization On The Internet,* Blackwell Publishers, US.
9. Lamsweerde, A. V. (2000) In *Proceedings of the 22nd international conference on Software engineering*, pp. Proceedings of the 22nd international conference on Software engineering.
10. Larkin, J. and Simon, H. A. (1987) Why a Diagram Is (Sometimes) Worth Ten Thousand Words.*Cognitive Science,* **11,** 65-69.
11. Luiijf, E. and Klaver, M. H. A. (2004) In *IEEE International Conference on Systems, Man and Cybernetics*The Hague, The Netherlands.
12. Mendonça, D., Lee, E. E. and W.A., W. (2004) In *IEEE International Conference on Systems, Man and Cybernetics*The Hague, The Netherlands.
13. Nielson, J. (1992) The Usability Engineering Life Cycle.*IEEE Computer***,** 12-22.
14. Nielson, J. (1995) Multimedia and Hypertext: The internet and Beyond In *MA: AP Professional*.
15. North, C. and Shneiderman, B. (2000) In *Conference Proc. Advanced Visual Interfaces*ACM, New York.
16. Peerenboom, J., Fischer, R. and Whitfield, R. (2001) Recovering from Disruptions of Interdependent Critical Infrastructures.Presentation to the CRIS/DRM/IIIT/NSF Workshop on "Mitigating the Vulnerability of Critical Infrastructures to Catastrophic Failures," Alexandria, VA.
17. Redwine, S. T. and Riddle, W. E. (1985) In Proceedings of the 8th International Conference on Software Engineering, pp. 189-200.
18. Ryder, B. G. and Soffa, M. L. (2003) Influences on the Design of Exception Handling.ACM SIGSOFT Project on the Impact of Software Engineering Research on Programming Language Design, **38,** 16-22.
19. Simon, H. A. and Hayes, J. R. (1976) The understanding process: problem isomorphs.*Cognitive Pychology,* **8,** 165-190.
20. Sommerville, I. (2001) *Software Engineering,* Pearson Education Limited.
21. Spoerri, A. (1999) InfoCrystal: a visual tool for information retrieval In *Readings in Information Visualization - Using Vision to Think*Morgan Kaufmann Publishers.
22. Sutcliffe, A. (2003) Multimedia user interface design In *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications* Lawrence Erlbaum Associates, Inc., pp. 245-262
23. Sutcliffe, A. G. (1999) User-centered design for multimedia applications.*IEEE Conference on Multimedia Computing and Systems, Florence,* **1,** 116-123.
24. U. S.-Canada Power System Outage Task Force (2004) In Final Report on the August 14th, 2003 Blackout in the United States and Canada: Causes and Recommendations.
25. U.S. General Accounting Office (2001) In Critical Infrastructure Protection Washington, D.C.
26. Vandenbosch, B. and Huff, S. (1997) Searching and Scanning: How Executives Obtain Information from Executive Information Systems.*MIS Quarterly***,** 81-105.
27. Wallace, W. A., Mendonça, D., Lee, E., Mitchell, J. and Chow, J. (2003) Managing Disruptions to Critical Infrastructure Interdependencies in the Context of the 2001 World Trade Center Attack In *Beyond September 11th: An Account of Post-Disaster Research*(Ed, Monday, J. L.) Natural Hazards Research and Applications Information Center, Boulder,CO, pp. 165-198.
28. Ware, C. (2000) Information Visualization: Perception for Design, Morgan Kaufmann publishers.