An Ontology-Based Distributed Whiteboard to Determine Legal Responses to Online Cyber Attacks

Leisheng Peng[1], Duminda Wijesekera[1], Thomas C. Wingfield[2], James B. Michael[3]

[1] Deptartment of Information and Software Engineering, George Mason University, Fairfax, VA 22030, USA

[2] The Potomac Institute for Policy Studies, 901 North Stuart Street, Suite 200, Arlington, VA 22203, USA

[3] Department of Computer Science, Naval Postgraduate School, Monterey, CA 93943, USA

Abstract. Today's cyber attacks come from many Internet and legal domains, requiring a coordinated swift and legitimate response. Consequently, determining the legality of a response requires a coordinated consensual legal argument that weaves legal sub-arguments from all participating domains. Doing so as a precursor for forensic analysis is to provide legitimacy to the process. We describe a tool that can be used to weave such a legal argument using the WWW securely.

Our tool is a legal whiteboard that allows participating group of attorneys to meet in Cyberspace in real time and construct a legal argument graphically by using a decision tree. A tree constructed this way and verified to hold anticipated legal challenges can then be used to guide forensic experts and law enforcement personnel during their active responses and off-line examinations.

In our tool the group of attorneys that construct the legal argument elects a leader (say the super builder) that permits (through access control) the group to construct a decision tree that, when populated by actual parameters of a cyber incident will output a decision. During the course of the construction, all participating attorneys can construct sub-parts of the arguments that can be substantiated with relevant legal documents from their own legal domains. Because diverse legal domains use different nomenclatures, we provide the capability to index and search legal documents using a complex International legal Ontology that goes beyond the traditional NeuxsLexus like legal databases. This Ontology itself can be created using the tool from remote locations. Once the sub arguments are made, they are submitted to the master builder through a ticketing mechanism that has the final authority to approve and synchronize the sub-trees to become the final decision tree with all its attached legal documents. Our tool has been fine tuned with numerous interviews with practicing attorneys in the subject area of cyber crime.

1 Introduction

When a malefactor intrudes into a computer system, the owner of that system—whether a private individual protecting personal property, a corporation securing its assets, or a government defending its interests—needs to know something about the

Peng L., Wijesekera D., C. Wingfield T. and B. Michael J. (2006).
An Ontology-Based Distributed Whiteboard to Determine Legal Responses to Online Cyber Attacks.
In *Proceedings of the 4th International Workshop on Security in Information Systems*, pages 232-245

malefactor in order to develop both a lawful and an effective response to the intrusion. Cyber intrusions may be characterized in one or more of three legal regimes: law enforcement, intelligence collection, and military operations. Furthermore, intrusions can occur across a number of jurisdictional boundaries, building complex conflict-of-laws questions into such attacks. Applying a one-size-fits-all response, such as always terminating all interaction with the intruder or always responding in kind, can be an ineffective or worse, illegal, response. For instance, terminating interaction with an intruder could prevent the seizure of evidence for criminal prosecution, collection of information for counterintelligence purposes, or counter-targeting for a military response [9]. By responding in kind, the defender may violate domestic or international law, or, in the case of a government actor, inadvertently escalate to the level of a use of force or even an armed attack. Furthermore, an inappropriately calibrated response may contravene the customary rules of war accepted as authoritative law by the United States—distinction, necessity, proportionality, and chivalry.

The general problem we address in this paper is that of providing defenders with sufficient information in order to make informed decisions when formulating responses to the actions of intruders. Specifically, we describe a tool that serves as an automated aid for determining the legal regime under which a cyber intrusion can be categorized, with documentation to support the building of a legal brief. Our tool is built on the premise that owners and their agents of affected computing resources want to defend their computer systems without violating domestic or international law.

Both the frequency and intensity of attacks in cyberspace can be high, affording little time for research and thoughtful consideration before the cyber intrusion (whether a crime, intelligence operation, or an attack) is over. Similarly, what may initially appear to be a minor intrusion or misuse of a computer system may ultimately result in damage to or destruction of property, or even human injury or loss of life. In either case, the defender must be prepared to respond to such attacks with operational plans and mechanisms for real-time information collection already in place; that is, the defender needs to tighten his Observe-Orient-Decide-Act (OODA) loop in order to gain a competitive advantage over the intruder [7].

Legal preparation is an essential element in this equation. Against opponents who disregard any laws which are not immediately and effectively punitive, the default response of inadequately counseled operators is to forego otherwise lawful and effective defensive strategies. In other words, the vast legal gray area that exists today operates in favor of the intruder—a form of asymmetry between the attacker and the defender. A clearer and timelier picture of the operational legalities of the situation would provide the defender with more, rather than fewer, intrusion-response options. At this stage in our research, several caveats are in order. First, the present tool is illustrative of the concept, and is not intended to be employed operationally at this point. The questions and answers have an artificially academic clarity, which derives from top-down reasoning of broad questions to narrow circumstances. Second, the decision-tree format no longer defines the state-of-the art in expert systems, but it does do the following: presents the core concept clearly; provides a framework with

law ontology that clarifies transparent assembly of resources that support legal analyses; enables the use of a distributed whiteboard for multi-level builders working on one decision tree concurrently without conflicting with each other; and lays a foundation for more elaborate logical structures (such as totality-of-the-circumstances analyses for future operational employment). Third, the inevitable anomalies which will arise in its development (*i.e.*, requiring an early legal determination of whether or not the intruder is a US person will almost certainly conflict with the operational reality of discovering key facts late in the game) serve to highlight conflicts and *lacunae* in the law. The degree to which the most operationally useful flow of legal questions fails to meet real-world requirements is the degree to which the law or technology must change. Fourth, and finally, this tool will be developed in alignment with international law, but numerous questions (especially in the law enforcement and intelligence collection realms) will never rise to the level of state versus state legal determinations. Where national and international law appear to conflict, that tension will be made explicit and thus clarified for resolution.

The remainder of this paper is organized as follows. Section 2 describes the details of our application requirements. Section 3 describes the software design of our toolkit. Section 4 explains the functionalities of the decision support system through an example. Section 5 describes related work and Section 6 concludes the paper.

2 Legal Requirements

As stated, our objective is to enable timely, effective responses to cyber incidents through legal means and the supporting documentation. In doing so, we are guided by the process of an attorney interviewing his client, establishing the legally operative facts of the case. Our larger objective is to make this a primary global requirement for responding to incidents in a timely manner. In order to reason about response alternatives, we first need a model of the domestic and international law governing cyber intrusions, one for computers to execute without the human in the loop and at high speed, and a second requiring human decision making at considerably lower speed. Our proposal for this model is a customizable decision tree of legally relevant questions. The computer decision tree will be hardwired for independent execution after meeting clearly discernable, objectively verifiable criteria. This layer of reasoning can be undertaken independent of human intervention, and thus can operate at the speed at which computers interact. The human decision tree will obviously operate far more slowly, and will proceed at the speed of human thought. This tree will present alternative paths requiring deliberate reasoning, and will be equipped with preselected sources to assist the attorney in deciding each of the gray-area judgments requiring human reflection and creativity. To do this, it will be necessary to assemble a comprehensive selection of legal sources and append them to each decision point. It will be vital, for speed and clarity, to include no more sources at any given decision point than would be required to answer the question at hand.

During in-depth discussions with experienced attorneys to identify and refine requirements for this system, ontology of how to categorize international and domestic legal sources emerges as an extremely useful and urgent-needed methodology to organize and retrieve the tremendous number of cyber legal documents. Any two countries may have very different expressions of very similar underlying legal principles. Identifying the core concepts and processes becomes the entry point of forming the legal ontology.

Our tool provides a way to categorize all cyber legal source documents through a general abstracted legal ontology. This general ontology can categorize any cyber sources, and store them in the system database for specific decision tree analytical support or more general thematic research. The ontology contains the principal identifiers of legal sources, such as title, catalog ID, author and credits, keywords, abstract, document type, and nationality. All legal documents may be broadly categorized as constitutional, legislative (statutes), executive (regulations), judiciary (cases), and international. These five categories must be further subdivided into primary (the case or statute itself) and secondary (analytic and synthetic commentary, such as law review articles, or briefs on file). These ten categories are sufficient to contain any legal source needed to address any given question. Furthermore, each source would have to be presented at four levels of abstraction, for the proper balance of speed and depth:

Citation: A legal footnote.
Précis: A sentence or paragraph paraphrasing what the source has to say about the question at hand.
Excerpt: Direct quotes from the source which are on point.
Document: The complete law review article, statute, or case.
Beyond these, each source has more optional details for categorization following appendix A.

This general information would be distilled into a specific research question in three media: **a tree map**, which provides a complete visualized tree structure with integrated relationships among tree nodes. This would give a very clear logic flow and the whole decision tree architecture; **an audit trail**, providing a record of each question asked and each answer chosen; and **a brief builder**, which would augment the audit trail with those portions of the sources selected by the reviewing attorney to support his answer to the question. This would, in effect, be the first draft of a legal brief supporting the selected course of action.

This decision tree, and its supporting sources, may be constructed using an open source methodology, allowing law students, practitioners, and scholars scattered across the world to collaborate on its construction and refinement. With the process architecture (described below) in place, the trees will be available to selected legal academics for analysis and improvement. Designing such a legal analysis tool for a comprehensive tracking system will be of great benefit to the cyber-legal community, because it will require the analysis and distillation of the entire field into the simplest possible framework for implementation.

This system will take the form of a set of predefined sequential questions when an actor's behavior indicates he or she may be intruding into, misusing, or attacking a computer system. To simplify the logic employed, in the prototype, each question has only *yes* and *no* answers. A deferent question will follow each *yes* or *no* answer to continue the analysis. Then attorneys and their clients would follow a complete logical path to reach a transparently reasoned legal conclusion. A third option, *don't know*, allows the user to view the legal resources to evaluate the immediate question, and make a principled decision to proceed forward with a *yes* or *no* answer. As mentioned earlier, these resources are arrayed under legal ontology from appendix A with ten categories (constitutional, legislative, executive, judicial, and international, each at a primary and a secondary level), and each source may be accessed at any one of four levels of abstraction (citation, précis, quotation, and full source).

This system will operate on two levels: for users following previously constructed analyses, and for *builders*, assembling and testing the analyses to be provided to the operational community. Users are attorneys responsible for providing operational legal advice to law enforcement, intelligence community, or military officials. These users follow a decision tree, answering a sequence of questions carefully crafted to identify and record the legally operative facts of the incident. This decision support tool will produce a logical legal analysis, supported by the legal resources selected by the user. Builders are authorized academic and practicing attorneys and some computer network technicians.

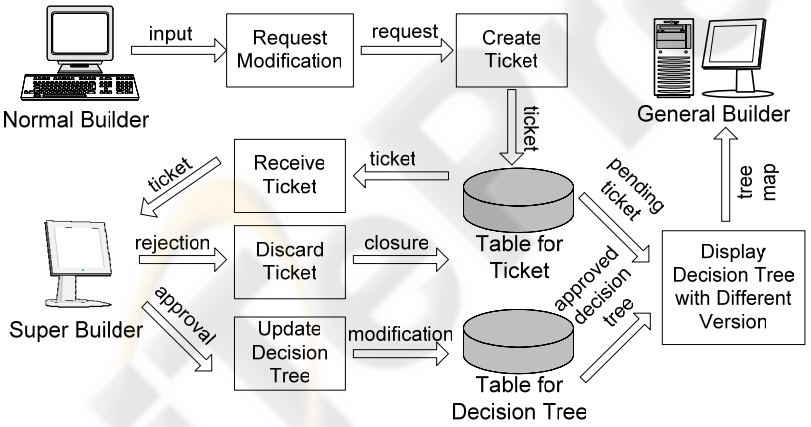

Fig. 1. Data flow diagram for distributed whiteboard.

There are three types of builders: *normal builder*, *super builder* and *administrator*. *Normal builders* may add and subtract branches from the decision tree and the resources available at each decision point. These changes, however, will not take effect immediately. A change-request ticket will be created when *normal builders* try to build a new tree or modify an existing tree. Then system will send these change-request tickets to authorized *super builders* who have the right to approve or reject the proposed modification. After *super builders* make these decisions, the approved changes will take effect immediately, and approved and disapproved change-request

tickets will be closed by the system. Although multiple *normal builders* may work on a decision tree and propose changes to the same tree branch at the same time, *super builders* still have full control of the decision trees and are able to avoid any conflicts during creating and maintaining the substance of the decision support tool. The detail data flow is shown in Figure 1. *Administrators*, who maintain the authorizations of all builders, not only can build or maintain decision tree directly, but also have the privileges to add and delete a builder, or modify the builder's login name, password, and the builder types (normal, super or admin).

The tree map is a visualized decision tree with all tree branches and tree nodes. Tree branches show the *yes* or *no* links between questions, and tree nodes present the questions with supported legal analysis resource. Current online *builders* may see who else is online at the same time, what changes he or she is making, and what the most recent approved decision tree is. By clicking the online builders' names, *builders* could compare their own tree maps with others tree maps or the approved decision tree maps instantly. All the tree maps, either changed by multiple builders or approved by *super builders*, will be displayed in the separate windows. So, for one decision tree, all types of *builders* can view the differences among those multi-version tree maps. The visualization of decision trees will not only avoid unnecessary redundancy or conflict before *normal builders* create change-request tickets, but also assist *super builders* in making the best decisions and guiding the decision tree modification to the correct direction.

3 System Design

This prototype is designed to be an open-source, Web-enabled decision support tool that provides legal reasoning Web services. Multiple clients may access the Web server (the system) via Web browsers, such as Internet Explorer (IE) or Netscape. The communication language between clients and the Web server is HTML exposed within Java Server Pages (JSP). A Java engine (Java 2 Software Development Kit, J2SDK) is used to compile the JSP pages to Java class files that stream HyperText Markup Language (HTML) to Web clients and communicate with a mySQL database through JConnector technology. Figure 2 shows the system architecture. Compared to client-server applications, our tool takes full advantages from the multi-tier design:

- Clients may remotely and concurrently access the system, sharing one knowledge base.
- The architecture is extensible, because it is built using the Java 2 Enterprise Edition (J2EE) service framework, with quick deployment times and minimal maintenance efforts in mind. Moreover, the system can be extended to use RDF, OWL, RuleML or JESS as needed.
- The system is easily manageable, because some clients are allowed to change the knowledge base line while other clients can only access built-in scenarios.

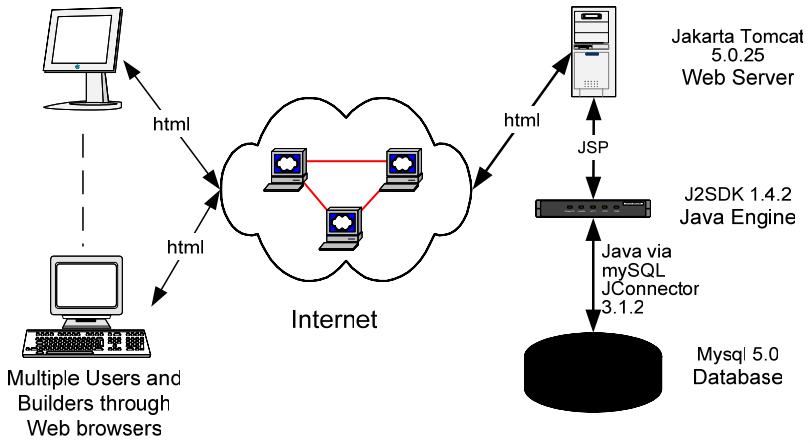

Fig. 2. System architecture.

Each client (a.k.a., actor in the lexicon of software engineering) is a builder or user with his or her own separate applications that share one database and file system. The user's main functionalities are: answering questions, making a decision, viewing (*e.g.,* audit trial, legal brief, or tree map), or searching and displaying pertinent legal documents under the legal ontology. The builder's main functionalities are: adding/deleting trees/decisions/questions, linking decisions/resources, loading resource under legal ontology (Appendix A), viewing current online builders and their tree maps, approving tickets by super builder, and administrating builder's group and login information.

4 System Functionality through an Example

This section describes the functionality of the tool by constructing an example decision tree to determine the answer to the legal question *Are we at war?* as shown in Figure 3. Again, there are three types of builders: *normal builder, super builder,* and *administrator.* All builders may access the system to build a decision tree via a Web browser after a correct login. Based on the unique login ID, the system will automatically identify the logged in builder's type and provide different functionalities according to the pre-authorized rights.

As a first step, the builder creates a new tree "*Applicable legal regime."* Then, the builder adds three possible decisions to this new tree. The builder then inserts multiple questions and links the appropriate follow-up questions or decisions to each. The builder must specify the parent question to which the new question is to be linked; that is, the builder should design the system so that a *yes* or *no* answer to a previous question is linked to a new question posed to a user. Because decision trees can be complex, our system is designed to offer the builder flexibility. For example, the builder can input the system's decisions and questions without having to enter the links when specifying them. After that the builder can use menu options to link the

decisions and questions. This way a complete decision tree can be constructed in multiple ways as shown in Figure 5 and Figure 7. When the decision tree needs to be modified because of the changes of law or some other reasons, our system provides sufficient administrative functionalities also, like adding, deleting, re-linking.

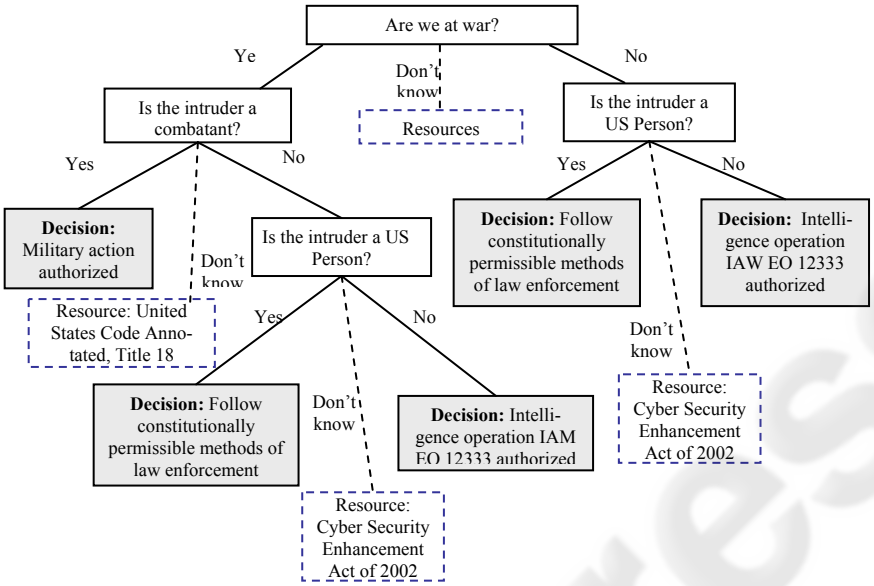

Fig. 3. Example decision tree.

When *normal builders* want to make a change for one decision tree or create a new decision tree, they only can send the change-request tickets which need to be approved by *super builders*. So *super builders* have the big vision and full control of decision tree while one or more *normal builders* are working on one decision tree. The ticketing approving functionality is shown as Figure 8. The system has a source searching engine as shown in Figure 4 to locate the original legal document. The search criteria could be very simple: just search the user's input by *keyword, title, author, citation, précis,* or *excerpt*. Users could also add *document type* and *country* into search criteria. The details of the search criteria for this search engine are shown in Figure 4.

Another important type of functionality in the tool is the distributed whiteboard. The system can provide a list of all online builders, wherever they are located. By clicking the hyper linked online user IDs, system will display the different tree maps in separate windows, as shown in Figure 7. For the same decision tree, different builders may have different change-request tickets to modify that tree. The tree maps displayed by the system will show the approved part of the decision tree, with the pending part of the decision tree highlighted in orange. Builders can compare the changes with the original tree map and with different versions from other builders. By moving the mouse over each decision node on the tree map, the question for each

node will appear automatically. By clicking on one of the nodes with supported resources, the tool will list all available resources for that particular question.

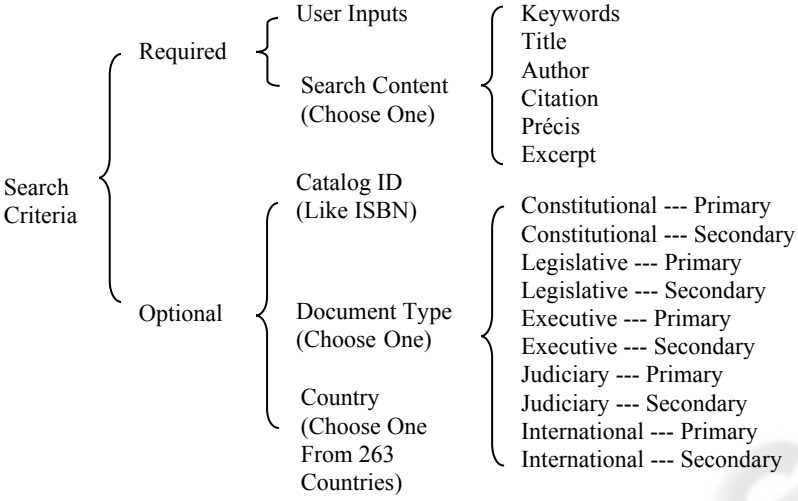

Fig. 4. Search criteria for legal source search engine.

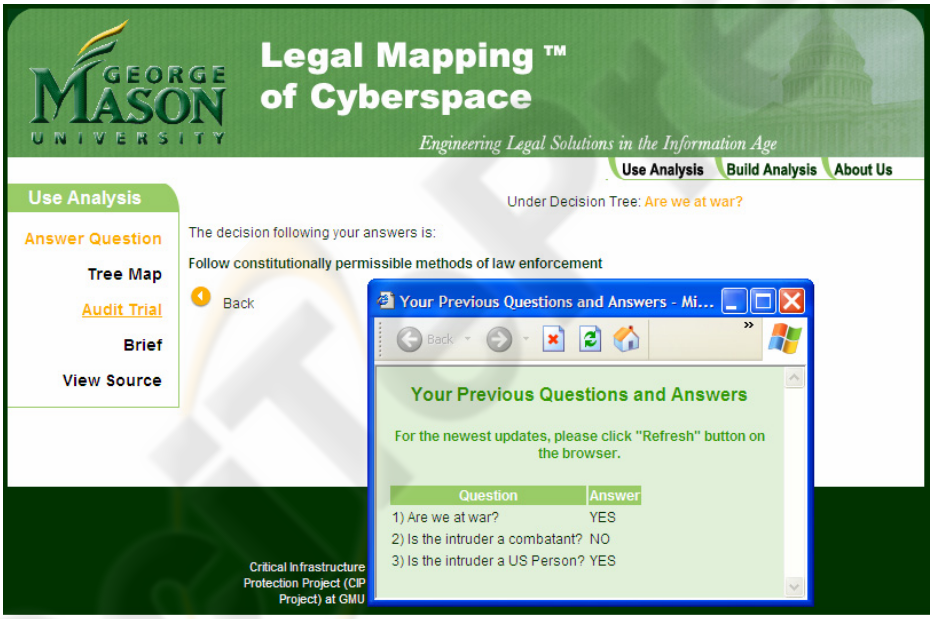

Fig. 5. A decision rendered by a completed tree.

Those resources for supporting analysis are categorized by the legal ontology from appendix A. In this example, the question *"Is the intruder a combatant?"* has a resource titled as "United States Code Annotated, Title 18", and the question *"Is the*

intruder a US Person?" has "Cyber Security Enhancement Act of 2002" as its resource. The Table 1 shows the differences between those two resources after being categorized by the legal ontology.

Table 1. Examples for legal ontology.

Categorization Field	Resource for Question *"Is the intruder a combatant?"*	Resource for Question *"Is the intruder a US Person?"*
Title	Cyber Security Enhancement Act of 2002	United states code annotated, title 18
Catalog ID	H.R. 5710	18 USC 121
Author & Credits	US Congress	US Congress
Keywords	Cyber security, sentencing, privacy rights, critical infrastructure, emergency disclosure, good faith exception, illegal devices, provider assistance.	stored communications, electronic communications, transactional records access, voluntary disclosure
Citation	Cyber Security Enhancement Act of 2002, H.R. 5710	Stored wire and electronic communications and transactional records access.
Type	Legislative	Legislative
Nationality	U.S.	US
Last Revision Date	2002	2001
Format	digital	digital
Location	http://www.cybercrime.gov/homeland_CSEA.htm	http://www.cybercrime.gov/ECPA2701_2712.htm
…	**…**	**…**

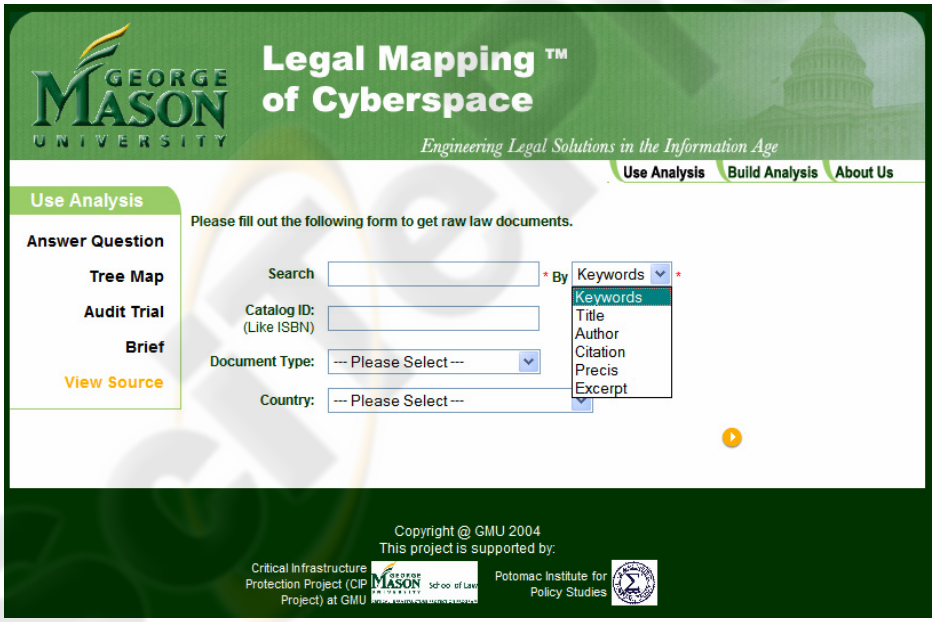

Fig. 6. Search criteria for source searching engine.

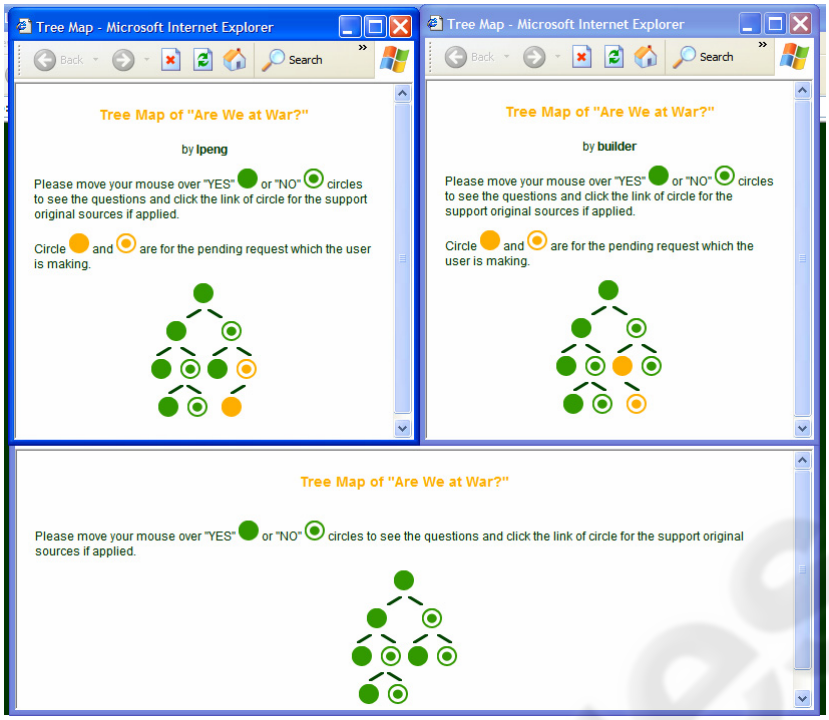

Fig. 7. Tree map comparison.

Fig. 8. Change-request ticket approval.

5 Related Work

Although there have been numerous academic attempts to elicit a logical structure from the legal decision-making process, none is in widespread use with practicing attorneys. The proprietary databases of Westlaw and LexisNexis, searchable by document category and Boolean keyword strings, are the most frequently consulted by attorneys. Both have an impressive number of cases, briefs, law review articles, and related documents [4,5,8], but neither is intended to provide direct assistance with the formulation or execution of a legal analysis. Furthermore, there are numerous free sites accessible via the Internet—mostly maintained by universities—that have large, searchable databases. Like their commercial analogs, they provide quick and reliable access to the documents themselves, but are not designed to assist in the legal analysis per se. The University of Minnesota's Human Rights Library is an excellent example, and is the source of UN Charter text provided in one of our examples [10].

Capturing legal knowledge and enabling legal discourse is both procedurally and substantively challenging because laws and their interpretation change over time. There are several legal reasoning support tools, such as [11, 12, 13], used primarily to hone law students' analytical skills. Others are geared for methodology or ontology research [1, 6, 13]. Only a few of these tools are comprehensive Web-based tools used for general legal reasoning [2, 14, 15], and therefore are not specific to one area of law. Duguid [3] and Zeleznikow [16] have developed Web-based tools to clarify the application of divorce law. In contrast, when fully developed, our tool may be used not only in training law students and cyberspace technicians, but will also provide real-time legal support for responding to cyber intrusions. Being both Web-based and open-source increases the usability, extensibility, maintainability, and capacity for stepwise enhancement of the tool.

6 Conclusion

People responsible for responding to cyber attacks can benefit from using a decision support tool to determine what options for response are available within the applicable legal frameworks. In order to address this need, we developed a decision-tree-based tool that leads investigators and attorneys through a series of questions that assist them in building legal briefs against cyber intruders. The decisions in a tree are to be constructed by attorneys who are well versed in a specific area of law: they construct a tree of sequentially ordered questions that guide the user through to an actionable recommendation for response (*i.e.*, the answer presented at a terminal leaf in the tree). In addition, our toolkit stores the relevant information within well-accepted legal categories (see Appendix A) at four levels of detail (citation, précis, excerpt, entire document) necessary to build a legal brief.

References

1. Gangemi A., A. Prisco, Sagri, G. M.-T., Steve, and D. Tiscornia. Some ontological tools to support legal regulatory, in Proceedings of the Workshop on Regulatory Ontologies and the Modeling of Complaint Regulations (Catania, Italy, Nov. 2003), Springer, Lecture Notes in Computer Science, pages 607-620, Catania, Italy, 2003.
2. K. Curran and L. Higgins. A legal information retrieval system. Journal of Information, Law and Technology, 3, 2000.
3. S. Duguid, L. Edwards, and J. Kingston. A Web-based decision support system for divorce lawyers, J. of Law, Computers & Technology, 15:265-280, 2001.
4. C. Hafner. Legal reasoning models, in International Encyclopedia of the Social and Behavioral Sciences, Elsevier Science Publishers, Amsterdam, 2001.
5. C. Hafner, and D. Berman. The role of context in case-based legal reasoning: teleological, temporal and procedural, J. Artificial Intelligence and Law 10 :19-64, 2002.
6. M. J. J. Hall, A. Stranieri, and J. Zeleznikow. A strategy for evaluating Web-based decision support systems. In Proceeding of the Sixth East-European Conference Advances in Data Information Systems, Bratislava, Slovakia, 2002.
7. P. E. M Huygen. Use of Bayesian belief networks in legal reasoning. In Proceeding of the Seventeenth British and and Irish Legal Education Technology Association Conference, Amsterdam, 2002.
8. E. Katsh and J Rifkin. Online Dispute Resolution: Resolving Conflicts in Cyberspace. Jossey-Bass, San Francisco, CA, 2001.
9. J. B. Michael. On the response policy of software decoys: Conducting software-based deception in the cyber battlespace. In In Proceedings of the Twenty-sixth Annual Computer Software and Applications Conference, pages 957-962, 2002.
10. J. B. Michael, and T. C. Wingfield. Lawful cyber decoy policy. In Gritzalis, S. D. C., di Vimercati, P. Samarati, and S. Katsikas, editors. Security and Privacy in the Age of Uncertainty. Pages 483-488. Kluwer Academic Publishers, 2003.
11. A. J. Muntjewerff. Automated training of legal reasoning, in Pre-proceedings Ninth British & Irish Legal Education Technology Association Conf. pages 51-58, Warwick, England, Apr. 1994.
12. A. J. Muntjewerff, A. Jordaans, R. Hoekstra, and R. Leenes. Case analysis and storage environment (case). JURIX, 2002.
13. V. Randall, OnLine Academic Assistance for Law Students, http://academic.udayton.edu/legaled/online/.
14. A. Stranieri, J. Yearwood, and J. Zeleznikow. Tools for placing legal decision support systems on the World Wide Web. In Proceedings of the Eighth International Conference on Artificial Intelligence and Law, pages 206-214, Sr. Louis, Missouri, May 2001. ACM.
15. A. Stranieri, and J. Zeleznikow. Tools for intelligent decision support system development in the legal domain, in Proceedings of Twelfth IEEE International Conference on Tools with Artificial Intelligence, IEEE pages 186-189, Vancouver, BC, 2000.
16. J. Zeleznikow. Using Web-based legal decision support systems to improve access to justice. Journal of Information and Communication Technology Law, 2002

Appendix A. Legal Source Categorization

1. Title = main label of the statute.
2. Catalog ID = publisher or provider's catalog number. It may be an ISBN.

3. Author and Credits = the author denotes the person(s) or organization(s) responsible for the creation of the statute; credits denotes contributors in addition to the author(s). Examples of authors may include an official body such as a legislature or agency who actually passed a law or regulation; while contributor may include a specific lawmaker's name who played a key role in moving the law or regulation forward.
4. Keywords = a list of key words.
5. Abstract = the main points of the statute. System searches can be made on words in the abstract. Four levels of abstraction: citation, précis, excerpt, and source (entire document).
6. Type = specifies the type in the broadest terms, such as constitutional, legislative (statutes), executive (regulations), judiciary (cases), and international. The type has subdivision of primary or secondary.
7. Nationality = country
8. Metadata Version = a designation of the legal database Version. This is made up of an organization designation and a numeric indicator of the version.
9. Length = number of characters or bytes.
10. New or old.
11. Expiration Date = when a statute may no longer be used.
12. Last Revision Date = the date on which the statute was last modified.
13. Format = the technically defined format of the statute, for instance, the digital format. Format may also define a non-digital format, such as a book or videotape.
14. Location - typically the URL(s) through which the container can be retrieved, either directly or through an index. Other addressing schemes can accomplish this objective.
15. User Rights = what a user can do with a statute –copyright law; permission rules.