How to Reuse Knowledge about Forensic Investigations

By

Danilo Bruschi, Mattia Monga, Lorenzo Martignoni

From the proceedings of

The Digital Forensic Research Conference

DFRWS 2004 USA

Baltimore, MD (Aug 11th - 13th)

How to Reuse Knowledge about Forensic Investigations

Danilo Bruschi, Mattia Monga
Università degli Studi di Milano
Dip. di Informatica e Comunicazione
Via Comelico 39 – 20135 Milano, Italy
{*bruschi,monga*}*@dico.unimi.it*

Lorenzo Martignoni
Università degli Studi di Milano Bicocca
Dip. di Informatica Sistemistica e Comunicazione
Via Bicocca degli Arcimboldi 8 – 20126 Milano, Italy
martignlo@disco.unimib.it

> These problems should be solved with that geometrical precision, which the mist of sophistry, the seduction of eloquence, and the timidity of doubt, are unable to resist.
>
> – Cesare Beccaria, *Of Crimes and Punishments,* XI

Abstract

When detectives perform investigations they manage a huge amount of information, they make use of specialized skills and analyze a wide knowledge base of evidence. Most of the work is not explicitly recorded and this hurdles external reviews and training. In this paper we propose a model able to organize forensic knowledge in a reusable way. Thus, past experience may be used to train new personnel, to foster knowledge sharing among detective communities and to expose collected information to quality assessment by third parties.

1 Introduction

From detectives' viewpoint, a crime involving computers is no more than another criminal case. They surely still believe in some form of the *Locard's Exchange Principle* [6] stating that "when two objects come into contact, a mutual exchange of matter will take

place between them". In fact, even in the cyber-world, every activity is likely to produce a modification in the system on which it is performed (for example, modifications in the file system, or– at least– modifications in the main memory of the computational device). Moreover, alterations in the information realm *must* have some correspondence within the physical world to express their actual criminal nature. Thus, every action may eventually produce an evidence, i.e., a piece of information known or assumed as a fact, whose relevance to further inferences has been established by investigators [10].

However, when evidence is *digital* in nature, special consideration is needed, since the smallest evidence change (a bit) may provoke the largest error in its interpretation. Thus,

- evidence might be easily and voluntarily erased;
- evidence might be easily and voluntarily forged (i.e., false evidence might be created);
- evidence might be altered accidentally by daily activities (i.e., the everyday use of a system might damage evidence);
- evidence at different abstraction layers, has different meanings and properties (e.g., an `html` document may be considered formatted text, or a sequence of `ASCII` characters, or a set of blocks in the file system structure) [2].

The traces left by the dullest script-kiddy could disappear as a result of normal system activity and, vice versa, the most meticulous hacker could leave on a system plenty of tracks of her or his presence. Therefore, a good detective must perfectly know *what* to look for, *where* to look and *how* to look while limiting the possibility of evidence compromise to a minimum. However, in order to achieve these results, skilled and experienced personnel is needed. As a matter of fact, crimes present *common patterns* that could be exploited to ease the work of investigators, but this knowledge is often tacit, only partially shared with colleagues, and mainly disorganized. Our work aims at providing a methodology for archiving, retrieving, and reasoning about forensic knowledge, in order to incrementally improve the skills and the work of a team of detectives.

Every investigation starts with a preliminary analysis of the crime notification (*notitia criminis*) which leads to the formulation of some initial hypotheses that drive the evidence discovery process. Hypotheses state something about the state of world in which the crime took place and they must to be verified (or falsified) in order to ascertain the true modalities of the action. We claim that hypotheses frameworks (i.e., the hypotheses space and their relations) are mostly reusable among similar cases and even among different cases that share similar parts. Thus, in this paper we propose an approach that, firstly, makes explicit the hypotheses space and the verification process and, secondly, since an explicit investigative process can be recorded, enables an easier sharing of experiences. Moreover, a clear expression of hypotheses and verification process makes possible a rational assessment of the quality of information collected. In other words, we aim at providing to the detective community both a systematic approach and a software tool able to:

- produce reusable forensic knowledge to be used as support during investigations;
- organize past experience to foster knowledge sharing among forensic experts;
- record collected information in a way that ease quality assessment.

The paper is organized as follows: Section 2 presents our approach to the whole investigative process and how it can be analyzed and decomposed. Section 3 formalizes our model while Section 4 describes its application to a real scenario. Section 5 discusses related work and finally Section 6 draws some conclusions.

2 Conceptual framework

2.1 The investigative process

A generic investigative process may be summarized by the schema depicted in Figure 1. After a preliminary analysis of the case, detectives:

1. formulate hypotheses on the state of the world that caused the case,
2. collect evidence on the basis of these hypotheses,
3. correlate actual evidence with hypotheses,
4. adjust hypotheses, and repeat the process until the consistency state of the knowledge about the case is sensibly high.

Finally, the case is interpreted by formulating a possible scenario that is able to explain the case. In this paper we address only the grey part of the picture, by proposing a structured way of thinking that helps detectives in managing its intrinsic complexity. Our model supports them while they start from (possibly recurrent) hypotheses and they try to prove their veracity; our approach guarantees that the collected information will be structured in such a way that makes it easier to understand, analyze, and reuse. However, we do not aim at providing any support neither for initial hypotheses formulation nor for their effective interpretation: activities that we believe it is much more productive to leave under the control of both human ingenuity and responsibility.

2.2 Managing the complexity of the real world

The real world is an inexhaustible source of complexity. Even simple cases may hide a bunch of subtleties and intricacies. This is especially true when computers and digital information are involved. Digital systems are designed by leveraging on a number of *virtual machines,* that provide a hierarchy of abstraction levels. Thus, users of a given virtual machine may safely forget about the underlying layers. Obliviousness of the internal layers of the "computational onion" is what save our sanity when we, for example, copy a document on a floppy disk, ignoring most of the technicalities of the file systems and devices. However, as far as investigations are concerned we should not ignore any

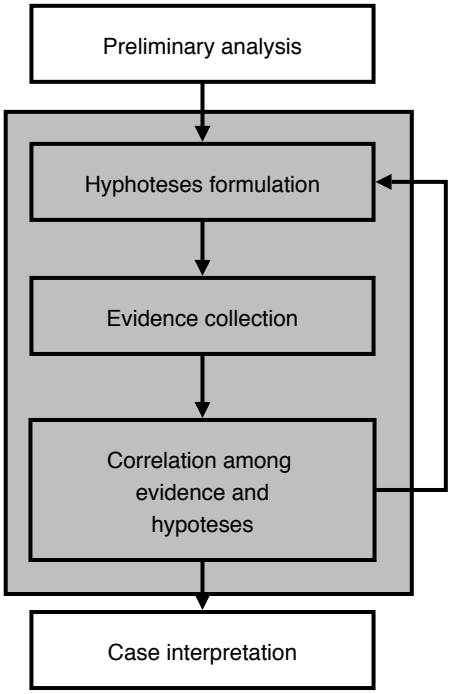

Figure 1: Investigation process

of the virtual machines involved, since everyone could be both a source of information and misinterpretation. Consider an apparently trivial example in which an investigation starts from the following hypothesis:

> H: Email account `user@domain`, registered by user Alice, has been used to send a harmful message M to user Bob. Alice was the author of M and its sender.

A demonstration of H that supports Alice's guiltiness and is able to resist in a penal trial, requires at least a complete analysis of the harmful message, a smart search of traces on systems owned by Alice, a careful study of SMTP logs, etc. If a phone line were involved in the transmission, data referring to it could be used to increase the confidence in the hypothesis. Instead, if no information about possible intrusions on Alice's machines were collected, our trust in her guiltiness should be modified to take into account that we cannot guarantee her system was either safe or owned by an unknown attacker.

How to cope with this complexity? Even a very simple case turns out in a very complicate investigation. In order to deal with such a complexity we find it very efficient to refer to the basis of *reductionism,* introduced by Renè Descartes. Whenever a problem is too complex, one should split it, resolve the simpler problems, and compose the final solution. Descartes summarized his method in four precepts [5]:

1. *Evidence:* nothing that is not clear and evident can be accepted.

2. *Analysis:* a problem that cannot be faced all at once should be decomposed in easier parts.

3. *Synthesis:* a decomposed problem has to be recomposed, but only after every part has been verified through detailed observations and considerations.

4. *Enumeration:* the whole process has to be reviewed to evaluate the soundness and completeness of the generalizations involved. Moreover, a careful revision is needed to ascertain the absence of errors and misinterpretations.

This conceptual framework drove us while we worked on our approach, since only a *clear and distinct* argumentation provides the bases for reusable knowledge.

2.2.1 Evidence

We need some facts, observations, real things to argument in favor or against a hypothesis. The belief in something must be demonstrated providing tangible data; data that can be collected, observed in order to draw conclusions. Nothing can be assumed neither true nor false when no observation was performed.

It is worth marking that the meaning of a datum and its relevance are not constant. There is evidence that corroborate an argumentation, but, at the same time, they contradict others. Some data are important in a situation and not in others. Data that are not important and that would not modify the belief in an argument do not provide evidence in that specific case. Evidences are *context sensitive*.

2.2.2 Analysis

Complex arguments ought to be separated in small ones. Our initial hypothesis H can be decomposed in sub-hypotheses, whose verification is simpler; this decomposition will be denoted by:

$$H \rightsquigarrow H_1, H_2, H_3, \cdots, H_n$$

In most cases detectives do not know enough about the structure of the world in order to sincerely affirm that the symbol $\rightsquigarrow$ denotes an actual logical equivalence. This has to be regarded has the limiting (and lucky) case that is unlikely to present itself in something different from academic examples. For the same reason "," should not be in general interpreted as logical conjunctions or disjunctions, since the knowledge about the world and possible event correlation is hardly complete.

Applied to the above described example the analysis rule may produce:

H_1 Alice has sent message M from her computer C

H_2 `sendmail`, the mail transfer agent installed on C, has been configured to use `user@domain` as the `From:` header

H_3 when M was sent (T), C has been in use

H_4 when M was sent (T), C was connected to the Internet

...

$H \rightsquigarrow H_1, H_2, H_3, H_4$

This rule is applied recursively until the last hypotheses cannot be decomposed anymore, or their decomposition would be meaningless. The final result is a *chain of reasoning*, a network of hypotheses that step by step becomes more and more complex rooted in the initial hypothesis H.

It is worth emphasizing some points:

- the decomposition applicable to a given hypothesis is by no means unique. There are plenty of ways to apply the decomposition rule, and there is no objective metric to grade them. However, since the experience of detectives plays a great role in this decision, it is very convenient to be able to make it explicit and record it for future use and revision;
- the chain of reasoning is in general an acyclic graph, since the same sub-hypothesis might be generated by several super-hypothesis and no loops are permitted among super- and sub-hypotheses.

2.2.3 Synthesis

The synthesis is the recomposition of the partial solutions of the decomposed problem. In the context of forensic investigations solving a problem should be interpreted as "collecting information to prove or disprove the occurrence of an event in the real world". In other words, in order to be able to draw a conclusive assessment about a case, detectives need to find significant tests to evaluate the simplest hypotheses. They have to analyze the scene of the crime in order to find elements that may enable them to estimate their rational belief in hypotheses. In other words, detectives perform tests aimed at collecting data that are relevant (i.e., provide information about discrimination between a hypothesis and its negation) in the assessment of a given hypotheses. We denote the mapping between evidence collecting tests E_i and hypothesis H as

$$H \mapsto E_1, E_2, E_3, \cdots, E_n$$

Coming back to our example, after all the necessary decompositions have been applied, we choose every simplest hypothesis and link it with evidence collection activities that might be performed in the attempt to verify its validity. Let us work with H_3, "when M was sent (T), C has been in use":

E_1 check if there are files modified, created, deleted, accessed on time T

E_2 check if there are files that contain information about user activity (browser history, email-client recent file list, ...) on time T

E_3 check if there are files that contain information about system activity (events logs, applications logs, ...) on time T

...

$H \mapsto E_1, E_2, E_3$

The evaluation of each of these evidence will lead, if applicable, to a success or to a failure. A success will favour $H3$ credibility, a failure will discredit it.

Again, it is worth emphasizing some points:

- the set of tests applicable to a given hypothesis is by no means complete. However, since the experience of detectives plays a great role in proposing relevant tests, it is very convenient to be able to make them explicit and record them for future use and revision;
- in some cases detectives know by experience that a certain test is relevant to assess the validity of a hypothesis, but performing it could be infeasible for several reasons (lack of resources, for example). Nevertheless the synthesis of the case should take into account also this information, since it affects the quality of the assessment;
- the "disproof force" of a negative result in a performed test is in general non-conclusive as the "proof force" of a positive results.

Therefore, an automatically computed, bottom-up proof of the root hypothesis is out of our intentions. This would be possible only if people could assign numeric probabilities to each hypothesis, under the even more unrealistic assumption of a completeness of decomposition. Instead, the synthesized weight of evidence is the product of the experience of the synthesizer, however what form the basis of her or his judgment should be recorded and provided to the study of third parties as indicated by the fourth Descartes precept of enumeration.

2.2.4 Enumeration

Cross validation is an essential part of forensic science. Therefore, a crucial point of any investigation methodology is its ability to foster independent reviews. Our approach support inspections by archiving the investigation process. Its graphical representation (see Section 3 for details) eases cross checking of investigators' choices. Moreover, reuse of past decompositions and test lists limits human omissions and errors, driving even less skilled investigators during their work.

3 Formalization of the model

After having presented our model in a descriptive way, we now formalize it. The model can, by nature, be represented as a network, an acyclic directed graph whose nodes are hypotheses, evidence collecting tests are attached leave hypotheses, and edges represent

decomposition links and the first application of the synthesis rule (i.e. link between evidence and hypotheses). The network can be described by the tuple FG, *forensic graph*, as follows:

$$FG =< H, E, F_h, F_e, w >$$

where:

- H is the set of hypotheses
- E is the set of evidence collecting tests
- F_h is a decomposition relation ($F_h \subseteq H \times H$)
- $w \in \{?, +, -\}$ is the *weight of evidence*, it is used to describe if the evidence has been analyzed and if the evidence test was performed and if it corroborates or contradicts a hypotheses
- F_e is an association relation, the application of the synthesis rule ($F_e \subseteq H \times E \times w$)

This graph is used to represent all the knowledge acquired over the time. This is the sink where we go to look for something already done in the past. Hypotheses and evidence are expressed in natural language. The weight of evidence is meaningless in the general model, it became meaningful only when the model is instantiated (i.e. it is used to represent the inference network built to model a particular case).The instantiation of the model will lead to the construction of a new graph, that will use only a subset of the whole set of elements; it will be called *case graph*.

To better illustrate the inductive reasoning used to prove or disprove a hypotheses we decided to propose a graphical formalism too. It is depicted in figure 2. Hypotheses are represented by square, evidence collecting tests by circle and the weight of evidence by a label on the edge linking evidence to hypotheses. The input of the decomposition rule is the root node of the graph and its output is the whole graph composed only by square nodes. The synthesis rule accepts, in input, the output of the previous phase and outputs the graph mades by both square nodes and circle nodes. The detective tests each evidence and records how they modify the belief in the hypotheses they are linked to: the "+" symbol means corroboration, "−" means contradiction and "?" signifies that evidence was not collected or not applicable.

4 An example

During a chat session a user has been caught spreading an offensive picture. After a preliminary investigation Mr. Black felt under suspicion. He has been accused of guilty because the address used by the sender to transmit the images, was, at that moment, assigned to him.

In the preliminary phase the detective, starting from the file received and the address of the sender, comes to identify Mr. Black as the criminal. Mr. Black's computer has been seized for further analysis. We can formulate the root hypothesis and try to applied the method above described:

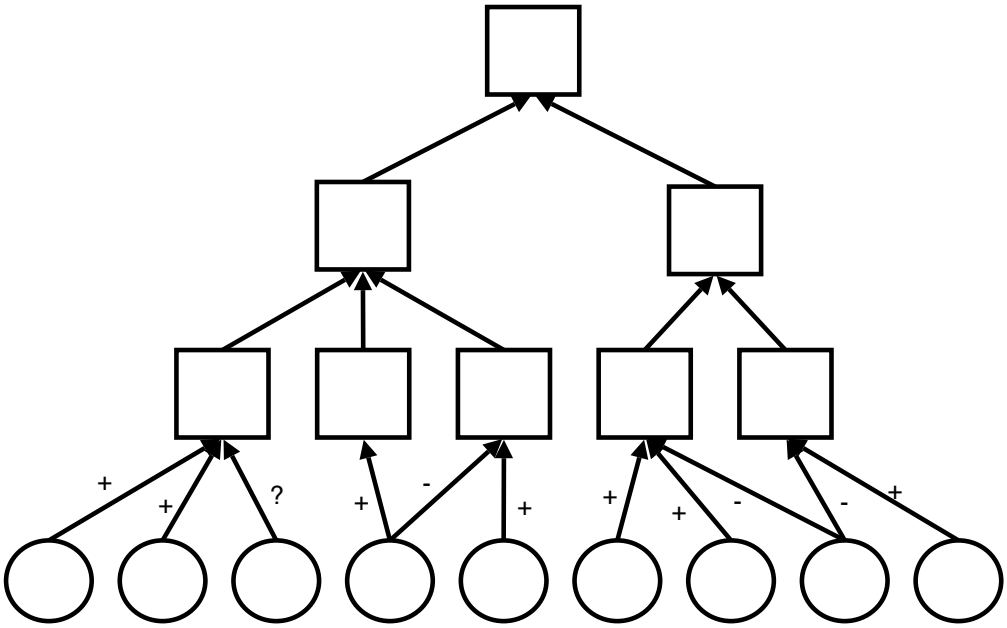

Figure 2: Graphical formalism adopted to represent case graph.

H: The defendant I (Mr. Black), on date D, sent the file F (child pornography contents), using the protocol P (irc).

to verify that Mr. Black voluntarily sent the picture. We must cover all possibilities, even that Mr. Black's computer was controlled remotely by a third person. In the past, people, firstly suspected of a computer related crime was then declared innocent because on their computer was installed malicious software probably used by someone else, remotely, to commit the crime [11].

The output of the decomposition rule and synthesis rule is the following:

H_1 I, on date D, possessed a copy of the file F

- $H_{1,1}$ I's system contained a file that corresponds exactly to the incriminated one (both metadata and content match)
  - E_1 F is found on I's system (both metadata and content match)
  - E_2 F was on I's system but it has been deleted (metadata matches, the recovered content corresponds to the original)
- $H_{1,2}$ I's system contained a file that corresponds only in part to the F (only metadata matches)
  - E_3 F is found on I's system (metadata matches but content does not)
  - E_4 F was on I's system but it has been deleted (metadata matches, the recovered content does not correspond to the original)

$H_{1,3}$ I's system contained only reference to F

E_5 references to F are found in user's history

E_6 references to F are found in user's documents

E_7 references to F are found in application's history

E_8 references to F are found in the swap space

H_2 I's system was used to send F

$H_{2,1}$ I's system was connected to the Internet on date D

E_9 The provider demonstrates that the sender address was assigned to I account, on date D

E_{10} The telephony operator demonstrates that I, on date D, was connected to the ISP

$H_{2,2}$ I's system had been used on date D

E_{11} There are files modified, created, deleted or accessed on D

E_{12} There are user files that contain information about user's activity (browser history, emails date, ...)

E_{13} There are system files that contain information about system activity (events logs, applications logs, ...)

$H_{2,3}$ I's system couldn't be spoofed by someone else

E_{14} TCP/IP spoofing attacks are hard to realize

E_{15} IRC client is configured to use the same nickname used by the sender of F

E_{16} IRC client is configured to connect to the same IRC server used by the sender of F

H_3 I voluntarily sent F

$H_{3,1}$ I's system contains a software that talks the IRC protocol

E_{15} IRC client is configured to use the same nickname used by the sender of F

E_{16} IRC client is configured to connect to the same IRC server used by the sender of F

E_{17} I's system contains a well known IRC client

E_{18} IRC log files indicates that the user uses regularly the client to connect to the chat

$H_{3,2}$ I's system had not been used by someone else to commit the crime

E_{19} I's system does not contain known backdoors

E_{20} I's system does not contain tools for remote control of the machine

E_{21} I's system contains only binaries and libraries known

It is important to note that we already see hypothesis $H_{2,2}$ in the example of section 2 and that evidence E_{15} and E_{16} are linked to more than one hypotheses.

The investigative process proceeds with the analysis of each evidence and marking their weight on the graph. There could be some evidence that can not be verified, for example for the lack of some information, of time or resources: a file protected with a password is found on the system and its decryption would require too much time. If the decryption is not accomplished, we can say nothing about its content so we have to tell that such kind of evidence can not be verified.

5 Related works

The methods appropriate to evaluate and defend the relevance and credibility of an evidence have been studied deeply by Schum [10]. He studied general properties of evidence and the structure of the chain of reasoning based on evidence. His research was mainly directed to the measure of the *weight of an evidence* using both a Bayesian and a Baconian approach. The former, based on Bayes' Theorem, requires a numerical assessment of the involved probabilities in order to evaluate the *inferential drag* of piece of evidence, thus we believe is not suitable to a juridical context. The latter (firstly analyzed and described by Cohen [3, 4], based on the seminal work by Bacon [1] in the XVII century) better suits our needs since the assessment of arguments is pursued by exploiting eliminative and variative induction, explicitly avoiding any assumption about numerical probabilities. The trust in a hypothesis is weighted only recording the results of the tests performed to invalidate it. We adopt a similar approach in which tests correspond to evidence finding.

The idea behind the case graph comes after the frameworks for argumentation mapping and critical thinking: the point is depicting in a diagram the argument that supports a thesis. The usefulness of this approach to model legal reasoning has been discussed in [7], comparing Wigmore's diagrams [14] with those produced by the Araucaria [8] tool. Wigmore's diagrams have been designed for constructing judicial arguments from a mass of evidence: they represent correlations among data and the chain of reasoning. Their syntax is quite complex and allows to distinguish among strong evidence, circumstantial evidence and testimonial evidence. Wigmore's charts can be viewed ad the predecessor of modern visual argumentation models. Araucaria, Reason!able [12, 13] and Athena [9] are tools to model inductive arguments.

Reason!Able graphically represents arguments with a tree, called argument map, where lines are used to indicate when evidence is related to a proposition and colors indicate favor or disfavor. No support to reuse of subtree is provided.

Athena provides the same functionalities of Reason!Able and furthermore it allows acyclic graphs instead of trees, where propositions may be linked at different levels.

Araucaria has a unique feature: supports the definition of a custom and reusable set of schemas that represent recurrent argumentations. The graphical notation of Araucaria, while quite different, has the same expressive power of the one used by Reason!Able and Athena.

6 Conclusion

Detectives who investigate computer crimes have to deal with several different criminal conduits and a chaotic mass of evidence data. It is unlikely that a single group of investigators has all the skills needed to interpret correctly all the evidence collected and understand all the intricacies of the different technologies involved in any non trivial case. Therefore, we believe that disciplined methodologies are needed and that the possibility of archiving forensic knowledge plays a crucial role in training and best practices dissemination.

Our approach, leveraging on argumentation theory, aims at:

- producing reusable knowledge, since forensic (sub-)graphs can be exploited to generate completely unrelated case graphs;
- structuring argumentation from evidence to prosecution hypotheses, since a graphical representation of the structure of the hypothesis space and the evidence support that was collected may convey, even at a glimpse, the global soundness and completeness of the information gathering;
- guiding less skilled detectives during evidence collection, since the highly specialized knowledge of experts in a field can be shared, thanks to its recording in a structured fashion.

We found our approach quite useful in a number of criminal cases in which we were involved. We were able to reuse some hypotheses and gathering techniques in different contexts. However, a critical point is hypotheses formulation. In fact, it is not clear how to compare and reuse hypotheses formulated with different scopes in mind, unrelated granularity, and, after all, expressed by using natural language. We are experimenting with a wider audience of law enforcement experts in order to find guidelines and best practice for hypotheses formulation. A tool consolidating our approach is under development: we plan to use it for collecting experience from other teams. Our final goal is a thorough study about how real world forensic investigations in computer crime are carried on, in order to tune our methodology.

References

[1] Francis Bacon. *The New Organon*. 1620.

[2] Brian Carrier. Defining digital forensic examination and analysis tool using abstraction layers. *International Journal of Digital Evidence*, 1(4), 2003.

[3] Johnatan L. Cohen. *The Implication of Induction*. Methuen and Co LTD, 1971.

[4] Johnatan L. Cohen. *The Probable and the Provable*. Methuen and Co LTD, 1977.

[5] Renè Descartes. *Discourse on the Method of Rightly Conducting the Reason*. 1637.

[6] E. Locard. The analysis of dust traces. part I. *American Journal of Police Science*, 1:276–298, 1930.

[7] Henry Prakken, Chris Reed, and Douglas Walton. Argumentation Schemes and Generalizations in Reasoning about Evidence. In *ICAIL*, pages 32–41, June 2003.

[8] Chris A. Reed and G.W.A. Rowe. Araucaria: Software for Puzzles in Argument Diagramming and XML. Technical report, Department of Applied Computing, University of Dundee, 2001.

[9] Bertil Rolf and Charlotte Magnusson. Developing the art of argumentation. a software approach. 2002.

[10] David A. Schum. *Evidential Foundation of Probabilistic Reasoning.* Wiley-Interscience, April 1994.

[11] John Schwartz. Acquitted man says virus put pornography on computer. *New York Times*, 11 August 2001.

[12] Tim van Gelder. How to improve critical thinking using educational technology. In *Proceedings of the 18th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education.* Melbourne: Biomedical Multimedia Unit, The University of Melbourne.

[13] Tim van Gelder. Enhancing deliberation through computer-supported argument visualization. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*, 2002.

[14] John H. Wigmore. *The Principles of Judicial Proof.* Boston: Little, Brown and Company, 1931.