Scalable Collaborative Risk Management Technology for Complex Critical Systems

Scott Campbell
Department of Computer Science
& Systems Analysis
Miami University
Oxford, OH
campbest@muohio.edu

Leigh Torgerson
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA
ltorgerson@jpl.nasa.gov

Scott Burleigh
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA
Scott.Burleigh@jpl.nasa.gov

Martin S. Feather
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA
martin.s.feather@jpl.nasa.gov

James D. Kiper
Department of Computer Science & Systems Analysis
Miami University
Oxford, OH
kiperjd@muohio.edu

Abstract

Collaboration among scientists in research projects and among engineers in the development of complex systems has become the rule rather than the exception. While this collaboration can occur inside a single organization or at a particular location the ubiquity of the Internet and information technology tools has permitted collaboration across long distances to become common. We describe here our project and plans to develop methods, software tools, and infrastructure tools to address challenges relating to geographically distributed software development. Specifically, this work is creating an infrastructure that supports applications working over distributed geographical and organizational domains and is using this infrastructure to develop a tool that supports project development using risk management and analysis techniques where the participants are not collocated.

This work builds upon successful risk assessment methods and tools. It is hoped that this system will aid scientists and engineers better perform the risk management process through enabling distributed participation. Additionally it is important to provide usable human-computer interfaces and data analysis tools so that meaningful decisions arise from the information.

1. Introduction

This project's focus is on creating tools and systems that support geographically and organizationally distributed software and specifically software aiding in the risk analysis process. To accomplish this goal we are combining two research areas. One research area involves methods and tools for managing objectives based upon risk analysis and the other involves researchers working on widely distributed information systems. The former are building upon successful research into tools and methods that support the development process through analyzing risk, cost and system functionality. These techniques are embodied by the *Defect Detection and Prevention (DDP) process* [3, 4, 5] which was developed by researchers at NASA and JPL. This process and tools are currently in use by engineers and aids them in identifying risks, analyzing possible minimization methods and comparing both the risk and minimizations against cost and project objectives. Currently this process is supported by a software tool that runs at a single location and requires people's physical presence. The DDP process has proven to be an effective tool but is limited by the single location constraint. Many potential developments are geographically and organizationally distributed. We are extending the DDP tool to work in widely distributed information systems so it can be applied to more development projects.

Rather than develop specific infrastructure methods just for the new DDP or limit DDP to existing functionality we are extending existing an existing middleware framework. It is anticipated that this extended framework which will support widely distributed information systems and multi-organizational projects will be applicable to projects beyond just the new DDP tool. Specifically we are extending JPL's Tramel communication system. Tramel provides inter-process communications via messaging and supports many platforms. We are enhancing Tramel so that it can operate over widely distributed systems involving communications between organizations with diverse security systems and requirements. One unique aspect of this research is its reliance on Delay Tolerant Network protocols (DTN). Most existing networks operate on a store/forward model but the data storage typically is expected to be on the order of fractions of a second.

Existing techniques fail if the delay extends to minutes or hours as might be the case of extremely long distances, extremely low bandwidth, or varied environments. These requirements can be common in environments in the NASA and JPL realms.

2. Project Objectives

This project's objectives fall into three general categories: developing software to support risk management analysis involving geographically distributed participants; developing infrastructure, tools, and techniques that efficiently operate between multiple locations, and validation of the risk management methods and tool implementations. There is a noted overlap in the objectives in the first two categories. This overlap is a consequence of bringing together two separate groups of researchers. One group's focus is on the development of risk management tools and the infrastructure, security and delay tolerant aspects are factors in the resulting environment. To the other group, these issues are the primary focus. We believe that this synergy of expertise and interest will help assure success as each group learns from and tests against the efforts of the other group.

The projects three objectives and associated sub-goals are:

1. Study the use of widely distributed information systems:
   - To provide a tool and method to help coordinate risk management in a distributed environment.
   - To assure security in this distributed tool.
   - To be tolerant of delays inherent on a distributed tool.
2. Design of widely distributed information systems:
   - To provide algorithms, tools, and protocols to allow such systems to be built in ways that assure security when systems are separated by multiple firewalls.
   - To assure delay tolerance in these algorithms, tools, and protocols.
3. Validation of an approach to distributed risk-based management:
   - To conduct controlled experiments to validate the improvement in design and analysis that results from use of the risk-based tool and method developed.

2.1 Study the use of widely distributed information systems.

We will study and create risk management techniques when used in a collaborative, decentralized, and asynchronous manner and as they are applied to intractably large and complex critical systems. Modern risk management tools are well suited to cost-effective analysis of risk in self-contained systems that can be fully understood by a small population of co-located experts. Increasingly, critical systems are being built that transcend these constraints: the modules of such systems may be broadly distributed, both geographically and organizationally, and the expertise for managing risk in those modules will typically be distributed in the same way. Society will benefit significantly from the enhanced reliability, capability, and availability of these systems.

2.2 Design of widely distributed information systems.

When designing a large-scale, widely distributed information processing system, several problems quickly crop up. Some of the nodes in the system may not be connected at all times, and the widespread use of firewalls and other protective techniques makes connectivity difficult. Additional difficulties include effectively distributing the processing tasks amongst multiple locations and performing processing tasks requiring data from multiple sources. One object of this research is to study these problems and, in the process, design and develop a specific and novel application to demonstrate the utility of the underlying network as it relates to solving problems in the development of large-scale distributed computational systems.

Such systems inherently have important security requirements. It is imperative to protect risk assessment data and ensure its integrity and confidentiality; otherwise

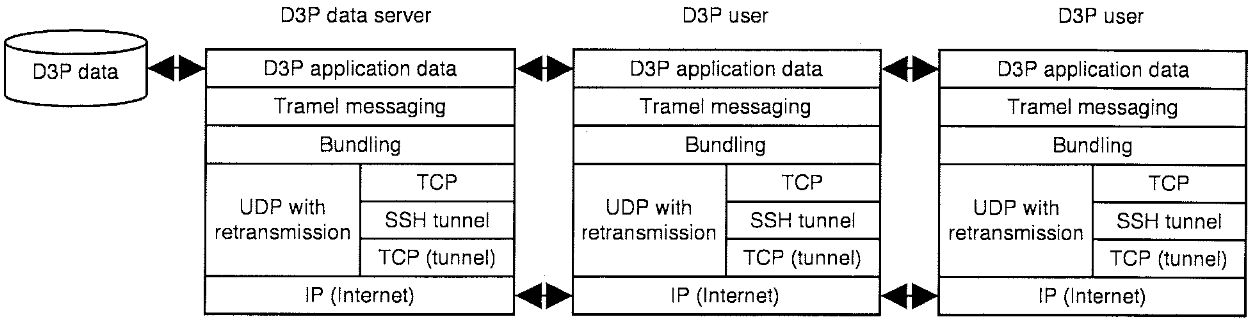

Figure 1 – Protocol Stack Architecture

it can be used against a project. Risk assessment inherently involves identifying and ranking vulnerabilities as well as listing steps taken to reduce or eliminate risks. Data integrity is important since decisions will be made based upon the stored data. Maintaining the data's confidentiality ensures that attackers do not use the data to identify the system's weak points. Unauthorized access to this data would give attackers a ranked list of potential vulnerabilities or allow them to direct attention away from potential risks. The security problems are compounded by the very nature of assessing large, complex projects. These projects are certain to involve parties with widely heterogeneous and distributed computing infrastructures.

Security emerges as a new concern for DDP as it is adapted to operate in a distributed, asynchronous environment. Currently DDP runs in discrete multidisciplinary sessions at a single location. The risk-based data collected from domain experts and associated tools resides on a single machine and is easily secured after the session ends. The proposed distributed risk analysis tool will store or make the data available from multiple locations and over an extended period of time.

It is necessary to segment the security tasks between the application and the infrastructure rather than place the majority of security work on either entity. Some tasks belong to the application and other tasks belong to the infrastructure. For example, only DDP or its operators can determine if a specific user should be allowed to update underlying assumptions to the model. It would be possible to have DDP handle the majority of security work but this would result in significant duplication of effort for other tools. It is best to rely upon the infrastructure to provide tools such as authentication, transmission integrity and privacy. Designing a tool so that it depends upon the infrastructure both relieves that tool of non-domain-specific tasks and effectively decomposes the tasks. We need research to determine how to best provide these functions in a delay tolerant environment.

2.3 Validation of an approach to distributed risk-based management.

It is critical that we evaluate the effectiveness of the new processes and tools. Specifically we will evaluate the efficacy of the project's software components and tools. Our plans include assessing the details of the tools, such as security and delay tolerance of our products. A more difficult task is assessing the effectiveness of using the tools for distributed design of complex systems. Thus, one of the specific objectives of this research is performing such an assessment.

3. Implementation Overview.

This project's goal is to create an environment supporting widely distributed systems and specifically the creation of a DDP tool that uses this environment. The new tool, the Distributed Defect Detection and Prevention (D^3P) extends the current DDP functionality thus allowing it to be used by a widely distributed population of risk analysts. To create the general infrastructure we are extending the JPL-developed messaging middleware system named "Tramel" so it can communicate between multiple sites and thus allow distributed D^3P data sources and users. Because of the sensitivity of this data and the likelihood that risk analysts will be working on computers that are behind their home institutions' firewalls, the network on which the Distributed DDP (D^3P) application operates will in effect be partitioned. To enable the efficient exchange of data across these partition boundaries, the Tramel-based system will rely on the capabilities of underlying Delay-Tolerant Networking (DTN) technology: DTN protocols can route Tramel messages bearing DDP data through firewalls without compromising security and without requiring ongoing and labor-intensive firewall engineering as the configuration of the application fabric changes.

Figure 1 shows the protocol stack envisioned for the D^3P system. Note that the D^3P software uses extensions to Tramel that, in turn, use a new bundling protocol that provides the necessary security and delay tolerance. Figure 2 shows the data flow operation of the D^3P integrated technology. This addresses the difficult problem providing secure communication through multiple firewalls.

Frameworks such as J2EE base much of their operation on the Remote Procedure Call model. Tramel supports the RPC model but also supports additional message exchange models such as asynchronous message passing and publish/subscribe that are better suited to very heterogeneous networks such as those we expect to encounter with large scale, complex systems. Traditional security mechanisms are not known to work, or are costly, when used between networks involving high latency, sporadic connectivity, or differing topology. For example, the establishment of a Secure Socket Layer (SSL) connection requires several TCP packet exchanges. For a very broadly distributed session of multi-user D^3P, the overhead of establishing and re-establishing large numbers of SSL connections between application endpoints might be intolerable. Tramel/DTN exchanges messages (bundles) between cooperating nodes using a variety of delivery modes that do not rely upon end-to-end TCP connections. Our hope is that this will enable applications to rapidly scale rapidly without compromising either performance or security.

This work will ensure that D^{3}P has the appropriate security components for operating in a distributed environment. We thereby will gain an understanding of the role and usefulness of D^{3}P for analysis and mitigation of risks in large, complex systems.

4. Relationship to present state of knowledge

This project has a clear relation to, and builds on, three on-going activities in which various project team members have been intimately involved. One is the existing DDP tool. [3, 4, 5, 8] The second major activity is the work on the messaging middleware system TRAMEL. [1, 10] This system will be extended in the manner described previously in this proposal. The third effort is that on delay tolerant networking. This is new work that is vital to the success of this effort. [2] Here are brief explanations of the current status of each of these.

DDP is a risk-based decision-making methodology conceived of and developed at JPL and NASA. DDP has been successfully used to aid in decision making in novel situations – for technology assessment, adoption, system design, development and operation. DDP is a process that uses a software tool. The DDP process involves stakeholders whose expertise spans the gamut of concerns in evolving a technology for mission utilization. The process guides the gathering of detailed, quantitative risk-based knowledge from those stakeholders. Custom software [3, 4] has been developed to support the DDP process.

DDP is a relatively sophisticated process that goes beyond the more traditional methods such as FMECA (Failure Modes Effects and Criticality Analysis). Its additional capabilities have proven useful to guide decision making in the early states of design, especially designs with concerns and applications that span multiple discipline areas. The DDP software is correspondingly sophisticated, and has been in development since 2000.

In the current version of DDP it is important that all the experts be together and able to simultaneously contribute to the gathering and analysis of this information as well as the decision making process. For example, when the resources available do not permit a satisfactory design solution, the representatives of the funding source for the project may be persuaded to augment the funding so as to allow a viable solution; alternately, the mission science community may be able to indicate where and how certain requirements can be weakened so as to permit a less demanding design that is feasible, a process we refer to as "descoping" [5].

4.2 The messaging middleware system: Tramel.

Tramel (Task Remote Asynchronous Message Exchange Layer), developed in the mid- to late 1990s at NASA's Jet Propulsion Laboratory is the basis for developing the widely distributed information platform. Tramel was integral to the Flight Systems test bed and has been an important development and test tool [7]. Tramel is a system for event management inter-task communication in distributed software.

Tramel insulates application code as much as possible from such inter-process communication details as connection establishment, transport protocol, and differences in processor architecture and operating system. Application software sessions are self-configuring at run time; the order in which processes begin participating in a session is immaterial. Tramel provides a built-in mechanism for linking reply messages, received asynchronously, to the contexts in which they are needed, enabling processes to converse in a pseudo-synchronous fashion without sacrificing parallel execution. It does so without requiring developers to master an OS-supported multithreading system; messages are by default processed sequentially, so access to the process's data is automatically serialized. Finally, Tramel supports an optional publish/subscribe communication model that further shields application code from having to understand the configuration or state of the distributed application at any time. In effect, each Tramel-speaking process (task) can plug itself into a data "grid", much as

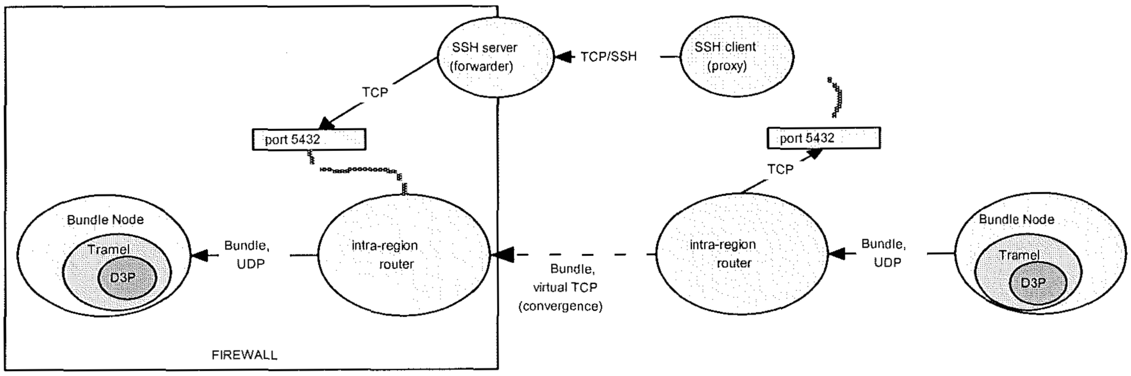

Figure 2 – Data flow with Tramel and DTN

producers and consumers of electric power – say, a hydroelectric plant and a kitchen toaster – plug into an electric power grid. A Tramel process can insert into such a data grid whatever data it produces, without having to know much about the consumer(s) of that data, and draw from the grid whatever data it requires without having to know much about the producer(s).

4.3 Delay Tolerant Networking.

We are leveraging work in Delay Tolerant Networking and its resulting network overlay capability to support development of widely distributed systems. Distributed communications inherently require the ability to exchange messages over dispersed environments. In widely distributed information systems the distributed communications typically will involve multiple organizations and the need to securely communicate between networks that each have differing security mechanisms. Organizations frequently have institutional and/or departmental firewalls to protect internal systems. It is necessary to communicate through these firewalls in methods that do not expose the organizations to undue risk and without compromising the data's integrity. This is a difficult problem since firewalls by their very nature inhibit the free flow of data to and from the end systems, and relaxing this inhibition to enable Tramel-based message exchange would require system administrator involvement in configuring the distributed application.

Involving system administrators in application configuration is not necessarily a worry in itself. However, much of the value of the proposed application architecture derives from its flexibility and tolerance of dynamic reconfiguration. If firewall adjustment were required every time the application configuration changed, either this flexibility would be lost or else the firewall engineering workload on system administrators would grow rapidly – increasing operations cost and/or the risk of firewall configuration error that might have institutional impact.

Our proposed approach is to operate the system over a virtual "overlay network" which exists above the protocol layer's transport layer. Each D^3P instance is a node on the network and communicates with other local nodes and with distant nodes via Tramel routers. The routers in this overlay network operate at the application layer of the TCP/IP protocol stack, and as such they can communicate among themselves using methods such as SSH secure "tunnels". University and corporate firewalls are typically configured to permit secure SSH traffic as standard procedure; wherever this is true, those firewalls would never need to be modified to enable the free flow of distributed risk management messages and dynamic reconfiguration metadata.

To support these nodes and routers we are building upon the Delay-Tolerant Networking (DTN) architecture work that has been developed over the past five years by engineers at JPL, MCI, Intel, The MITRE Corporation, and SPARTA, Inc. DTN protocols offers comprehensive support for operations on an abstract overlay network. By adapting Tramel to run over DTN's bundling protocol Tramel can securely route traffic between geographically sites without undue impact on institutional network security.

5. Technical Approach

This project has four components:

- Creation of an Internet-based distributed risk management tool to enable and facilitate design of complex systems in a widely distributed environment.
- Integration of a mature risk management tool with a messaging middleware system to produce this distributed risk management tool.
- Integration of the messaging middleware system with a new "overlay network" technology for improved security and scalability.
- Design of an experiment that will exercise distributed risk management technology to solve a large-scale risk management problem.

The Distributed Defect Detection and Prevention (D^3P) tool and related infrastructure that results from this work will be produced in an incremental fashion. An important component of this incremental process is the validation of each step to assure that it is still progressing toward project goals.

5.1 Validation

After completing each increment, we plan on conducting a set of experiments to validate the project goals. These include ensuring:

1. Validity of D^3P calculations.
2. Security of communications between sites.
3. Data store integrity and security.
4. Correct function of tool in the presence of delays and failures of network connections.
5. Usability of D^3P.
6. Ability of D^3P tool and process to effectively support engineering complex systems.

These sets of experiments will become increasingly complex as the tool increments gain functionality. Here we describe how each of these experiments will apply to each increment.

5.1.1 Validity of D^3P calculations. For each increment, the validity of these calculations will be determined by a careful and systematic comparison of results from the D^3P tool with those from the original DDP tool. There are several large collections of project data available since the original tool has had a considerable amount of usage at the Jet Propulsion Laboratory.

5.1.2 Security of communications between sites. The enhanced Tramel/DTN tools will provide a bundled message transmission architecture where messages flow between application nodes. It will be necessary to validate (both by review and experiment) that the applications and infrastructure protect messages' confidentiality, ensure messages' integrity, and validate their authenticity. There are unique challenges when using the power of a DTN where messages can successfully delayed and stored for long periods at intermediate machines. These messages must be kept confidential and their integrity maintained at multiple storage locations where system administrators do not have typical assurances of operating system security. Hence it will be up to the Tramel/DTN network to provide these assurances. It will thus be necessary to actively test Tramel/DTN's ability to secure these messages by actively attempting to subvert the network. Specific tests will be performed that attempt to maliciously modify, delete, change and inject messages such that applications are unaware of the attacks.

5.1.3 The D^{3}P data store will be changed to an as yet unspecified storage model. Regardless of the storage model we will perform tests to ensure that the data can only be accessed by properly authenticated users and that the users can only access the data in authorized roles. Additionally, testing of backup/recovery and site-recovery methods will be necessary. Site recovery differs from standard backup/recovery in that it is limited to a site recreating its data store from other stable/correct data stores.

5.1.4 Correct function of tool in the presence of delays and failures of network connections. To validate this property of each increment, we will inject delays into the network, and determine if this affects the correctness of results. We will be developing techniques to maintain data consistency and coherency of data between sites over such a network. Tests of these methods will include planned network outages, delays, and anomalous behavior such as "random" underlying message replication, reordering and delays.

5.1.5 Usability of D^{3}P. When each increment of the D^{3}P tool is complete, we will conduct a set of usability tests on the GUI. It is important that these tests be conducted on each increment so that any problems detected can be corrected in subsequent increments. The two primary approaches that we describe in our plan to evaluate the usability of DDP [8], and also plan to use here, are usability testing and cognitive walkthroughs. Usability testing involves having typical users performing prescribed tasks with the system or process being evaluated. Subjective (Likert scale satisfaction, for example) and objective (time to complete tasks, number of errors) data are collected and analyzed. As described in [9] this technique is an "information-processing model of human cognition One difference between these two techniques is that cognitive walkthroughs do not require user participation. Expert evaluators examine typical usage scenarios, explore possible user responses, and evaluate alternatives. This allows application of cognitive walkthroughs to be applied to incomplete interfaces, or in situation when users are not available for usability testing.

5.1.6 Ability of D^{3}P tool and process to effectively support engineering complex systems. This is the most difficult of the validation tasks since it is difficult to quantify whether one design is better than another, or whether one process has been more effective than another. However, we will attempt to validate these properties through a group of controlled experiments. We will recruit a group of engineers at two or three NASA sites to participate in a study. We will divide these engineers into two groups, matching them as closely as possible for background, education, and experience. We will find two moderate-sized projects at JPL (or elsewhere in NASA) that are at the point where they need a design study. (For example, one of these may be a study of a new technology.) It is important that both group consist of engineers at geographically separated sites. Group 1 will design project A in a traditional manner; then will design project B with D3P. Simultaneously, group 2 will design project A with D^{3}P, then design project B in a traditional manner. Upon completion, we will give a third group of experts a set of design criteria and ask them to evaluate the quality of all four projects. In addition, we will survey and interview the two participant groups to assess their opinions about the efficacy of D^{3}P. As a part of the experimental procedure we will collect data on productivity, hours spent, etc.

6. Conclusions

The development of a Distributed DDP using Tramel messaging over DTN will provide an easily extensible and adaptable state-of-the-art framework for many research and education projects. The concept of delay-tolerant networking and the use of standard messaging services will allow instrumentation, signal processing and data analysis to be readily distributed across any number of networks, with geographically and temporally distributed researchers. The D^{3}P application may itself be used in managing risk in multi-university collaborative project settings, and the new infrastructure will enable any number of distributed database, signal processing or data mining applications.

7. Acknowledgements

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration, and at Miami University, Oxford, Ohio. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government the Jet Propulsion Laboratory, the California Institute of Technology, or Miami University.

8. References

[1] C. K. Ames, S. Burleigh, B. Auernheimer "An Environment for Incremental Development of Distributed Extensible Asynchronous Real-Time Systems", *Proceedings of the 4th International Workshop on Parallel and Distributed Real-Time Systems*, 1996.

[2] Burleigh, Hooke, Torgerson, Fall, Cerf, Durst, Scott, Weiss, "Delay-Tolerant Networking: An Approach to Interplanetary Internet", *IEEE Communications Magazine*, June 2003, vol. 41 no. 6, pp 128-136.

[3] Cornford, S. L., M. S. Feather, et al. (2001). DDP – A tool for life-cycle risk management. *IEEE Aerospace Conference*, Big Sky, Montana, 2001.

[4] Feather, M.S., Cornford, S.L. Dunphy, J. & Hicks, K.A. (2002). A Quantitative Risk Model for Early Lifecycle Decision Making; *Proceedings of the Conference on Integrated Design and Process Technology*, Pasadena, California, June 2002. Society for Design and Process Science.

[5] Feather, M.S., S.L. Cornford & K.A. Hicks (2002) Descoping; *Proceedings of the 27th IEEE/NASA Software Engineering Workshop*, Greenbelt, Maryland, Dec 2002. IEEE Computer Society.

[6] M.S. Feather & S.L. Cornford. Quantitative Risk-based Requirements Reasoning; Requirements *Engineering Journal* (Springer); (8):4, 2003 pp 248-265.

[7] Mark L. James, Ed Baroth, Lee Mellinger, Han Park, Tim Stough; Integrated Virtual Test Bed for IVHM Systems on 2nd Generation RLV. *IEEE Aerospace Conference*, Big Sky, Montana, 2003.

[8] Kiper, James, Brent Auernheimer, and Martin Feather, Assessing Usability of a Risk-based Requirements and Design Tool, *Proceedings of the 7th IASTED International Conference on Software Engineering and Applications*; Marina del Rey, CA. Nov. 2003.

[9] Rieman, J., Franzke, M. & Redmiles, D. Usability Evaluation with the Cognitive Walkthrough. http://www.acm.org/sigchi/chi95/Electronic/documnts/tutors/jr_bdy.htm

[10] Ann E. Kelley Sobel, "Security Analysis of Tramel", *Proceedings of HICSS 32*, January 1999.