Technology Transition of Network Defense Visual Analytics:

Lessons Learned from Case Studies

Brianne F. O'Brien, Anita D'Amico, Mark E. Larkin
Secure Decisions Division
Applied Visions, Inc.
Northport, NY 11768 USA
{brianne.obrien, anita.damico, mark.larkin}@securedecisions.com

Abstract – **Despite more than a decade of significant government investment in network defense research and technology development, there have been relatively few successful transitions across the chasm between research and operational use. Prior work describes approaches to crossing the "valley of death" from the perspective of the government sponsor or independent tester. The researcher and developer's perspective offered in this paper adds to our understanding of the challenges faced and solutions applied to deployment of advanced technologies into operational environments. The paper describes lessons learned from recent transitions of two information assurance technologies – the VIAssist® netflow visualization tool and the MeerCAT® wireless vulnerability analysis tool – into operational use by the Department of Homeland Security (DHS) and the Department of Defense (DoD).**

Keywords – computer security, homeland security, network defense, technology readiness level, technology, transfer, technology transition

I. Introduction

Despite more than a decade of significant government investment in network defense research and technology development, there have been relatively few successful transitions across the chasm between research and operational use [1]. Often, successful transition is due to a dedicated program manager, and "opportunistic channels of demonstration, partnering, and occasional good fortune." [2] The limited work published on this topic has been largely from the perspective of government sponsors of research, who are appropriately concerned about return on the considerable investment made in research, or from independent evaluators of prototype technologies. [3] Other work has suggested the formation of transition integrated product teams (TIPTs) that provide a natural means for bringing key stakeholders together to identify and address transition issues. [4] This paper offers the perspective of researchers who have transitioned three cyber security visualization technologies from the laboratory into operational environments in the government and, to a lesser extent, industry. With each research-to-product transition we have applied the lessons learned from prior successes and failures. This paper shares several lessons learned from two recent case studies of moving up the Technology Readiness Level (TRL) [5] ladder from Level 1 to Level 9.

II. Case Studies

A. *VIAssist*

VIAssist is a visual analysis platform to help network security analysts protect their networks. It provides visual tools for the evaluation of network flow and security data [6]. VIAssist traversed the TRL ladder over six years of start and stop funding, during which proof of concept, prototype, and production versions were developed. The foundational requirements for VIAssist were developed during an Intelligence Advanced Research Projects Activity (IARPA) funded cognitive task analysis of cyber defenders [7]. The proof-of-concept and first prototypes were funded by IARPA and by Air Force Research Laboratory (AFRL). DHS Science and Technology (S&T) supported the enhancement of VIAssist into production-quality software and the build-up of a technical support infrastructure. AFRL is further expanding VIAssist capabilities to support Security Information and Event Management (SIEM) products within Air Force operational environments. VIAssist is now accredited for deployment at US-CERT.

B. *MeerCAT*

MeerCAT (Mobile Cyber Asset Tracks) is a set of integrated visualization tools to help cyber vulnerability analysts assess risks to critical infrastructure from wireless 802.11 threats. It is also used by penetration teams, for assessing vulnerabilities that can be exploited to gain access into targeted networks. Originally developed under a Defense Advanced Research Projects Agency (DARPA) funded Phase II Small Business Innovation Research (SBIR) contract, MeerCAT has entered the SBIR Phase III stage, with sustainment funding from Defense Information Systems Agency (DISA) and Naval Research Laboratory (NRL). It has been adopted and accredited for the DoD Information

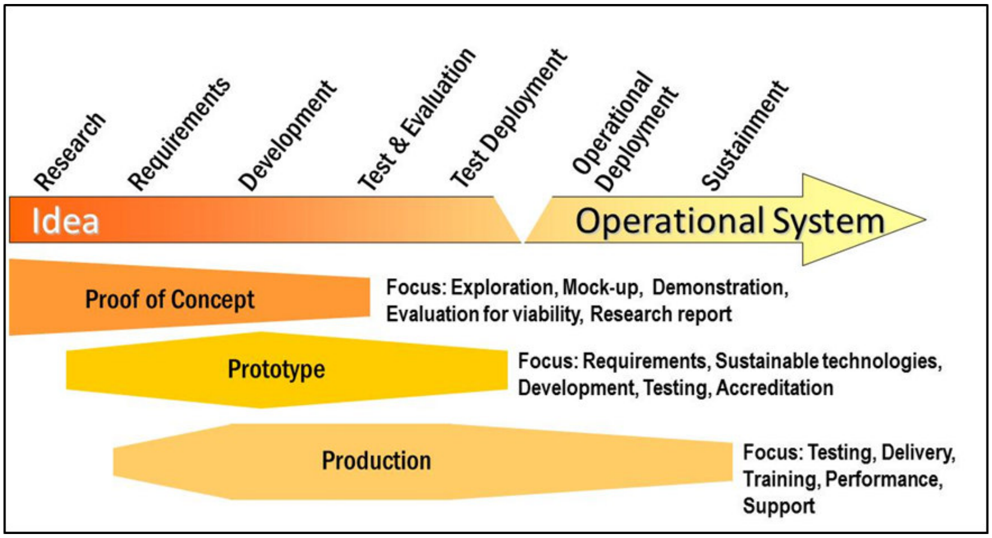
Research
Requirements
Development
Test & Evaluation
Test Deployment
Operational Deployment
Sustainment
Idea
Operational System
Proof of Concept
Focus: Exploration, Mock-up, Demonstration, Evaluation for viability, Research report
Prototype
Focus: Requirements, Sustainable technologies, Development, Testing, Accreditation
Production
Focus: Testing, Delivery, Training, Performance, Support

Figure 1. The path to transition is comprised of several discontinuous phases

Assurance Certification and Accreditation Process (DIACAP) as a component of the DoD Flying Squirrel wireless discovery suite. As part of that suite, MeerCAT has been downloaded more than 2,000 times by government users. MeerCAT is also available commercially, through subscription, individual perpetual licensing, and enterprise licensing.

III. Lessons Learned

There are many lessons that have been learned from this transition experience. We summarize them below, starting first with distinctions between project phases, and then proceeding from requirements through sustainment.

A. *Project Phases*

Lesson: Transition is a discontinuous process. Research programs often treat transition as though it is a smooth continuum from research to prototype to operational system. It is not. Different people, processes and technologies are needed in different phases.

As shown in Fig. 1 there are really three types of systems produced along the transition path: proof of concept, prototype, and production system. Each has different objectives and expends resources accordingly. The different shapes in Fig. 1 reflect the relative focus on the activities from Research through Sustainment. A research project (through TRL 3) produces a proof of concept that illustrates the viability of a solution. A prototyping project (through TRL 6) produces a requirements-driven working system that is subject to functional testing, demonstration and the initial stages of accreditation. The project that ends in an operational deployment expends considerably more time on testing than either of the prior phases, must be accreditation-worthy, and also supports end-user needs for instruction and support.

Lesson: Do not set false expectations by labeling a throw-away system a “prototype.” If you are developing a system to assess the viability of an approach or set of technologies, and do not think the system can withstand rigorous testing, refer to it as a “proof of concept.” We presented an early version of VIAssist to potential transition partners for initial review and feedback. While the fundamental user interface worked, the backend solution was less developed. We unfortunately referred to the immature version with an appealing UI as a prototype rather than a proof of concept. This led some potential transition partners to believe the VIAssist technology was more mature and closer to completion than reality. In fact, significant development effort and funding for a prototype version was still needed.

Lesson: Change project leadership as the effort moves through the three phases. The same project leaders that can successfully perform research and deliver a proof of concept are not well equipped to lead prototyping and production. A person comfortable with technology experimentation and tolerant of having some ideas fail is needed for TRLs 1 through 3. This same person may not be a good choice to lead prototype development where requirements documentation and tracking is needed, and where trade-offs are made between innovation, performance, and operational form fit. We have noted this in our own projects, and it has also been documented in Software Engineering Institute technology transition literature [8].

Lesson: When specifying requirements for and designing a prototype, also consider production and commercialization needs. The infrastructure and acquisition policies of the transition targets can affect choices of operating system, software, and hardware, and can influence trade-offs of cost versus performance. Our choice of Java for MeerCAT was driven in part by suitability to environment of our user champions. Our selection of two display surfaces in VIAssist was influenced by observations of multi-screen use in government security operations centers more than six years ago, when multiple screens were not ubiquitous.

B. Stakeholder Champions

Lesson: Regularly engage with at least two stakeholders that can serve as champions for your work from its earliest stages.[4] User champions will provide requirements, introduce you to other users, provide feedback on demonstrations, and serve as beta testers. Other forms of stakeholders can explain accreditation requirements, or assist with authorities to operate. Align yourself with at least two to mitigate the effects of losing one. Stay in touch even when your project is on hiatus, or engaged in a long development cycle.

Lesson: Provide some technology that will be of immediate value to stakeholders. Many research programs, by their very nature, focus on challenging, long-term needs. Stakeholders typically have a more immediate pain point that can be solved along with the longer-term need. When developing VIAssist we found that network defenders wanted a streamlined method for producing watch changeover briefings. We added an easy-to-use report builder to accompany VIAssist's sophisticated visual analytics and data querying technologies. This delivered the immediate relief of a report builder along with more advanced visualizations requiring some time investment to learn.

Lesson: Get testimonials from beta testers. Very happy beta testers, particularly in government, may be reluctant to provide testimonials to the success of a prototype because of approval hurdles. When possible, seek commercial beta testers who are apt to publicly share their views of the value of your work.

C. Requirements

Lesson: Establish distinct requirements for each phase of the transition. Requirements define what is to be built, what functionality it is to provide, and how it is to perform. Specific sets of requirements need to be defined for each phase of the project. Without proper requirements specification and clear completion criteria, projects can wander aimlessly without producing substantial results. They ultimately expend all funds with little hope of transition.

For the proof of concept phase, a minimal set of requirements is needed. Requirements for this phase must cover only the necessary functionality needed to demonstrate that the proposed approach is viable. It is critical at this phase that the requirements directly reflect functional needs. Proof-of-concept applications typically don't leave the developer or demonstration environment, so issues of performance, security, aesthetics and other nonfunctional requirements are of less importance than the direct functional requirements specified by the potential users. Once these requirements are fulfilled by the proof of concept application, then a determination can be made as to whether to move forward into a prototype phase.

During the prototype phase, an updated set of requirements can be defined to build a more functionally complete and demonstrable prototype application. Requirements for this phase would include the expanded functionality provided by the prototype as well as non-functional requirements pertaining to performance and stability since the prototype may be deployed for preliminary evaluation.

During the production phase, requirements are again updated to reflect a fully deployable, stable and maintainable application.

One of the challenges in these multiphase projects with potentially different stakeholders and funding sources is maintaining a consistent application development direction. Each phase is a balance of the initiatives of the product developers with the desires of the current stakeholders or end users.

Lesson: Develop requirements using stakeholder input. Requirements should be developed in collaboration with the stakeholder or immediate users of the proposed technology during each phase. Stakeholders may vary from phase to phase. For VIAssist, the requirements were identified up front during a Cognitive Task Analysis (CTA) preceding the development of the initial proof of concept.

D. Testing

It is important to formulate a robust test plan that addresses *what* will be tested and *how* it will be tested across multi-phased software development efforts. The testing rigor and test artifacts become more challenging as one moves from proof of concept through accreditation and production. Throughout all phases however, generating repeatable and documented test cases is an essential element in moving toward successful transition and accreditation.

Lesson: Do not skimp on testing; it's as important as development in the final prototype and production phases. While requirements define the expected functionality and behavior of an application, testing provides a means to validate the application against these requirements.

Lesson: Test early and often. Testing is not just an end of phase exercise. Testing should start early and be executed continuously through the development of the application. Early and frequent testing provides continuous feedback as to the health of the application throughout the development evolution. Should the addition of new or updated functionality adversely affect some aspect of the system, this will be detected immediately instead of at the end of the phase when the cause of the issue would be more difficult to identify.

Lesson: Create a test plan and tailor it to the phase. A test plan defines all aspects of testing to be done and includes the following:

- What is to be tested
- How it is to be tested
- When it is to be tested
- Types of tests to be performed
- Who is performing testing
- What resources are required for testing
- What test data is required for testing
- How are test results collected
- Specification of Test Scripts, Test Scenarios
- Traceability matrix matching tests to the requirements they validate

The process of creating the test plan itself is as important to development as the resulting test plan document. By forcing developers and testers to understand how the target application will be tested, they are less likely to put off addressing difficult issues that could stand in the way of a successful first deployment downstream.

Lesson: Use a test plan to clarify and consolidate expectations of stakeholders. The test plan is a collaborative tool to be developed with input and feedback from the stakeholders. By involving the stakeholders at this stage of development, it sets and clarifies expectations of the application.

Lesson: Obtain or create a test data set that will exercise and stress the system. Test data can be of two types: synthetic data, or real operational data. Synthetic data which can be generated by a custom utility application has the advantage of being well understood and can form the basis for much of the functional testing. In the case of VIAssist this meant we could generate data with particular patterns that would produce predictable graphing results. It also allowed for the generation of data to exercise the aggregation functionality of VIAssist, and to performance stress testing. Such data is indispensible for validating an application.

But synthetic data as described above cannot always tell you how an application will perform in a real-world operational environment. Even if your intent is to generate synthetic data that represents the real world, it is never clear if you have achieved this. Acquiring real operational data for Network Defense type applications is difficult. Organizations are reluctant to provide such sensitive data for obvious reasons. Ultimately, the only way to test an application with real work operational data may be in an operational setting outside the development or lab environment.

E. Demonstrations & Public Relations

Lesson: Have a compelling scenario-driven demonstration of your technology. This is essential for eliciting user feedback, and attracting potential transition sites.

Demonstrations typically focus on either features or on utility. Engineers and developers tend to favor the feature-based approach describing one feature at a time. This is adequate for demonstrating an early phase proof of concept, but insufficient for prototype and production phases. Potential users are more interested in the utility of the overall tool rather than in the individual features from which that utility derives. We have learned to craft demonstrations of our technologies around real-world scenarios that are instantly recognizable to potential users: scenarios that are representative of the sorts of problems they deal with every day. For example, when developing a demonstration of MeerCAT's wireless discovery visual analytics we collected wireless data by "wardriving" around a local shopping mall. The resulting demonstration illustrated the value of the technology against a background of commercial enterprises handling credit cards and medical facilities handling private data – something that all could relate to. We demonstrated a variety of the software's features, and discussed the technological underpinnings, as the scenario of the drive around the mall unfolded.

Lesson: Find or develop a demonstration-appropriate data set. Compelling technology demonstrations should be supported by a rich data set that illustrates key features and discriminators of the technology.

Lesson: Always have a demo. Throughout the development effort, a working demonstrable version of the software should always be available.

Lesson: Put your technology where your users can encounter it. Provide information sessions and demonstrations at conferences where users gather. While there, solicit feedback and additional requirements from subject matter experts, and identify beta testers and early adoption sites. Provide printed and electronic materials that describe the technology features and benefits, the needs the technology meets, and the users who are most apt to benefit from the use of the technology within their operational environments.

F. Certification & Accreditation

Lesson: Obtain transition partner requirements for attaining an Authority to Operate (ATO) during the prototype stage to allow ample time to comply with transition site security requirements. Performers targeting deployment must obtain an ATO that will permit the installation of the technology on a selected government network. Such ATOs are attainable through various formal processes, and support for these processes can absorb substantial resources in terms of calendar time and man-hours. ATOs are granted by Designated Approving Authorities (DAAs).

Requirements imposed by the DAAs will vary for certifying and accrediting application security. Government organizations are mandated, by Federal Information Security Management Act of 2002 (FISMA) regulations, to develop and implement programs that provide information security for the systems and information that support their organizational operations. Each government organization has some latitude to direct how FISMA compliance will be achieved. Depending on the organization, compliance with certain specifications may be required such as the National Institute of Standards and Technology (NIST) Risk Management Framework (RMF), DoD DIACAP requirements, or other organization-specific requirements.

G. Working with Your Transition Partner

Lesson: Establish a risk management plan with your transition partner to address factors that could impact the deployment schedule. A risk management plan quantifies the criticality of each risk in terms of time, budget and staffing, and offers corrective actions to be taken. Examples of risks are: delays in delivery of hardware and software, personnel turnover, delays in receiving security clearances. Maintaining a deployment schedule becomes more critical as the performer's period of performance closes in.

Lesson: Establish an alternate Test and Evaluation (T&E) site during the prototyping stage. It offers abatement of project risk if primary site scheduling delays are significant, and the secondary site can be leveraged as a test case for a trial deployment and preliminary evaluation.

H. Installation and Deployment

Lesson: Allocate more time than you, or your transition partner, initially estimate for installation and deployment. While the concept of installing an application at an operational site seems straightforward, there are numerous issues that can pop up and make this task more difficult than expected, such as improperly configured hardware, inappropriate credentials, and lack of access to transition site administrators.

Lesson: Have a deployment plan. A smooth installation requires deployment planning both by the developers and the deployment site. Develop a deployment plan and share it with the deployment site well in advance of the actual deployment event. A typical deployment plan includes the following:

1) Application minimum requirements: OS, platform, CPU, RAM, graphic capabilities, and any other specific requirements your application might have.
2) External Interfaces requirements: What external interfaces does your application require
3) External data sources: Does your application have specific data requirements? These should actually be worked out well ahead of time
4) Application name and version
5) Application installer
6) Application installer guide or instructions
7) Application configuration and setup details
8) Level of access needed to install your application or system

Lesson: Before leaving for the deployment, verify that necessary support personnel will be available to assist if required. This includes persons with administrative rights to the platform on which your application will be installed, and persons with administrative rights to the external resource you will be connecting to (such as a database).

I. Issues Specific to Government Deployments

Lesson: An Authority to Operate allows you to install your technology on the transition partner's network, but it doesn't allow for the technology to become fully operational. Be prepared to support a series of technical evaluations and administrative exercises to complete your deployment.

Deployment of VIAssist within our transition partner's operational environment was achieved through a series of incremental evaluations and documentation exercises that consumed a fair amount of calendar time to complete. This process was sequential in nature so commencing this approval process early in the transition activity was vital. Each step in the process brought VIAssist closer to fully operational capability. Initial evaluations consisted of an assessment of all VIAssist software components, libraries, and utilities that would be deployed on the transition partner network, followed by a stand-alone operational evaluation on a stand-alone test network. A subsequent proof-of-concept evaluation was required where VIAssist and other key technologies were further assessed within a more realistic test environment using sample data. Meanwhile, a Voluntary Product Accessibility Template (VPAT) needed to be prepared which entailed VIAssist evaluation from the perspective of 508 Compliance/usability for the impaired user. A Technology Insertion (TI) package was also generated and submitted to the government, which described the current and future costs associated with the purchase, upgrade, installation, integration, licensing, maintenance and training required for successful transition and sustainment. Approval of this package, in conjunction with previous evaluations, allowed VIAssist to be included on the DHS Technical Reference Model, a preferred vendors list. Additional approvals were required to allow the technology to be deployed onto operational workstations, using only canned data. A series of test cases were exercised against pre-defined transition partner technical requirements. Further approvals were required to allow VIAssist to access and use real organizational data after successful test case completion.

Lesson: Request the addition of a Contract Security classification Specification (DD-254) to your government contract, even if you don't think you'll need it.

The possibility always exists that a performer may be faced with gaining access to a classified facility, network, or data store even though your technology is unclassified. Entrance into the transition partner facility without a security clearance may be denied, or access can be severely restricted. This situation can impose a significant burden on performers, limiting them from participating in on-site integration, test, verification, remediation, and training activities. Within DoD and DHS venues, performers are encouraged to request a DD-254 rider to their performance contracts to avoid this potential barrier. It is the only authorized vehicle for conveying security classification guidance for use of classified security information.

J. Training

Lesson: Training is needed any time the application leaves the development environment and is exposed to new users who may evaluate it, not just at the time of final deployment.

Initial users can have significant influence on the first impressions and evaluations of the application. It is very important that they be trained in how to use the technology. Lack of training can result in a poor evaluation of the technology and become an impediment to the project transitioning to an operational environment.

Lesson: Develop training materials as soon as the first prototype is ready. Developing training earlier in a project provides the added value of understanding how to present the application and how the application presents itself to new users. Such insight is easily overlooked when the only ones who have been exposed to the application are the developers themselves.

K. Sustainment

Lesson: Establish a technical support infrastructure to include robust documentation and training, and professional grade utilities and Help Desk prior to deployment.

The technical support strategy implemented for VIAssist was to provide the end user with enough education and resources to enable them to effectively use the technology and produce results with a fair amount of ease. The depth and breadth of the user manual and self-study guides was expanded to improve user comprehension. Training materials were augmented to include hands-on exercises to engage users further, and to cover fundamental concepts and more advanced technical topics. Concepts for tuning or configuring other technologies, and data sources that interface with the transitioning technology, should be discussed with the goal of improving the quality of the data inputs to the system.

The Installation program supporting VIAssist was upgraded for ease of use, and the installation guide was improved to provide clear instructions. A Help Desk was established that provides technical support to end users via several different communication media, including a telephone call center with an automated call response chain and human support, and email inquiries that are responded to within less than one working day. A product knowledgebase website should be established that is easily accessible to end users, and initially robust enough to address FAQs and recommended actions in response to common error messages. Hot fixes and patches should be downloadable on demand from web or FTP sites.

Lesson: Technology sustainment becomes a dual responsibility and a cooperative effort between performer and the transition partner.

Once the technology is deployed, the next challenge is ensuring that the product is sustained by its continued use and its continued improvement. The performer must establish a plan for regularly scheduled product releases that not only contain bug fixes, but also enhancements and new features. A reasonable release cycle may consist of one "major" release and one "minor" release per year. A mechanism should be established for continuously engaging end users for feedback, in the form of an online trouble ticket system, user forums or blogs, and webinars, with a goal of continuously identifying new features and future developmental initiatives that will address user needs. From the acquisition agent's perspective, technology sustainment means providing funding to support technology maintenance, and continuous training for new employees and refreshers for others. It is important for financial officers to understand that funding for technology does not end when technology is purchased. Sustainment dollars must be factored in as new line items to facility budgets. Without this mindset, the product sustainment burden falls solely on the performer's shoulders.

L. *Contractual Obstacles to Transition*

Lesson: If commercialization is the goal, seek to remove obstacles that limit the market, such as federal export control restrictions. Determine if the commercializable technology falls under export control, and address it about one year before planned commercial release.

The Department of State International Traffic in Arms Regulations (ITAR), and the Department of Commerce's Export Administration Regulations (EAR) laws prohibit unlicensed export of information related to military and commercial technologies for reasons of national security and protection of trade to foreign countries. An assessment of the need for a license must be conducted, and is dependent upon the technology's characteristics, destination, end user, and end use [9]. The VIAssist evaluation process consumed significant calendar time; consequently it is important to begin this process early so that commercialization efforts are not impeded or delayed.

IV. CONCLUSIONS

Transitioning research results into an operational environment requires changes across many dimensions of a long-term program. Project leadership must shift from Principle Investigators focused on experimentation, to Project Engineers focused on production-quality technology. Testing changes from a focus on viability to one on robustness and scalability, and the resources provided for testing must increase accordingly. Relationships with transition partners also shift, from an informal relationship with stakeholder champions to a partnership in which both parties adhere to schedule and resource allocation commitments.

Making transition partners and future users comfortable is important for laying the groundwork for technology transition, and can take many forms: a scenario-driven demonstration, concise technology descriptions, ongoing training of anyone using the technology even in the early stages of a program, and user manuals.

Finally, as one moves closer to production and deployment, spend more resources on testing and deployment planning.

REFERENCES

[1] W. D. Maughan. "Crossing the 'Valley of Death': Transitioning research into commercial products - a personal perspective." In Proceedings of IEEE Symposium on Security and Privacy, 2010. pp.21-26.

[2] "A Roadmap for Cybersecurity Research," Department of Homeland Security, November 2009, page B2.

[3] S.C. Paladino and J.E. Fingerman, "Cybersecurity Technology Transition: A Practical Approach, in Conference for Homeland Security," 2009, CATCH '09, Cybersecurity Applications & Technology, 3-4 March 2009, pp. 325-330.

[4] J. A. Dobbins, "Planning for Technology Transition", Defense AT&L, March-April 2004, pp. 14-17.

[5] Homeland Security Institute, "Department of Homeland Security Science and Technology Readiness Level Calculator (ver 1.1), Final Report and User's Manual", 30 September 2009.

[6] A. D'Amico, J. R. Goodall, and J. K. Kopylec, "Wireless cyber assets discovery visualization," VizSec 2008: Proceedings of the 5th International Workshop on Visualization for Computer Security, Springer LNCS, 2008, pp.135-143.

[7] A. D'Amico, K. Whitley, D. Tesone, B. O'Brien, and E. Roth "Achieving cyber defense situational awareness: A cognitive task analysis of information assurance analysts." Proceedings of the Human Factors and Engineering Society Annual Meeting. September 27-29, 2005, pp. 229-233.

[8] E. Forrester, "A Lifecycle Approach to Technology Transition", Software Engineering Institute, News at SEI, September 1, 2003.

[9] U.S. Department of Commerce, Bureau of Industry and Security (BIS), Introduction to Commerce Department Export Controls, Updated May 5, 2003. http://www.bis.doc.gov/licensing/exportingbasics.htm