Identifying Commercial Off-the-Shelf (COTS) Product Risks: The COTS Usage Risk Evaluation

David J. Carney
Edwin J. Morris
Patrick R. H. Place

September 2003

TECHNICAL REPORT
CMU/SEI-2003-TR-023
ESC-TR-2003-023

Carnegie Mellon
Software Engineering Institute

Pittsburgh, PA 15213-3890

Identifying Commercial Off-the-Shelf (COTS) Product Risks: The COTS Usage Risk Evaluation

CMU/SEI-2003-TR-023
ESC-TR-2003-023

David J. Carney
Edwin J. Morris
Patrick R. H. Place

September 2003

Dynamic Systems Program
COTS-Based Systems Initiative

This report was prepared for the

SEI Joint Program Office
HQ ESC/DIB
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

Christos Scondras
Chief of Programs, XPK

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (http://www.sei.cmu.edu/publications/pubweb.html).

Table of Contents

List of Figures

List of Tables

Abstract

The expansion in use of commercial off-the-shelf (COTS) products has been accompanied by an increase in program failures. Many of these failures have been due to a lack of familiarity with the changed approach that COTS products demand. This report describes the development of an approach to reduce the number of program failures attributable to COTS software: the COTS Usage Risk Evaluation (CURE). The origin of CURE and an overview of the method, along with detail on the materials and mechanisms used in CURE, are provided. The CURE process is outlined and the results of the evaluations that have been conducted are summarized. Finally, possible future directions for CURE are explored.

1 Introduction

Over the past decade, the use of commercial off-the-shelf (COTS) products to implement significant portions of a software system has grown in both government and industry. The use of COTS products emphasizes buying commercial capabilities rather than building unique ones. Organizations that adopt a COTS-based systems approach generally expect either more rapid or less costly system construction. These organizations also hope to stay in step with the advances in technology that occur in the competitive marketplace. Government organizations are particularly encouraged to use COTS products by acquisition reform regulations.

The use of COTS products can indeed have a beneficial effect. For example, the National Aeronautics and Space Administration (NASA) successfully employed COTS products in re-engineering the Hubble Space Telescope Command and Control system [Pfarr 02]. The effort incorporated over 30 COTS and GOTS[1] software components and achieved improved capability while decreasing the use of custom code by 50 percent. The program was able to meet an aggressive schedule, delivering 10 major releases of the system in 48 months.

Unfortunately, for each highly successful case like the Hubble Command and Control system, there are other less successful projects whose shortcomings are often attributed to the use of COTS software products. The Software Engineering Institute (SEI[SM]), through its participation on Independent Technical Assessments (ITAs), has been called upon on numerous occasions to examine programs that have overrun their budgets and failed to deliver systems. In a very large percentage of these, COTS software has been a central issue in the failures. The SEI's experience is not unique. In a well-publicized example, Hershey Food Corporation's quarterly profits fell 19 percent in 1999, a drop largely attributed to an inability to get candy to Halloween and Christmas markets [Hayes 02]. It was widely believed that the cause of the delay was a newly-installed COTS Enterprise Resource Planning (ERP) system.[2]

This paper will report on the SEI's development of an approach to reduce the number of program failures attributable to COTS software: the COTS Usage Risk Evaluation (CURE). In Section 2 we describe the origin of CURE and provide an overview of the method. In Section 3 we provide detail on the materials and mechanisms used in CURE. In Section 4 we de-

[1] GOTS refers to government off-the-shelf components, which are available for incorporation into government-sponsored systems.

[SM] SEI is a service mark of Carnegie Mellon University.

[2] It should be noted that upgrades to the Hershey Foods system have gone far more smoothly, and with great benefit to the corporation.

scribe the CURE process. In Section 5 we summarize the results of the evaluations that we have conducted. In Section 6, we describe future steps to be taken.

2 Overview of CURE

In this section we provide a high-level description of CURE: why it was developed, a general outline of the method, and some guidelines on the kinds of organizations that might benefit from it. A full description of the process and techniques of CURE will be deferred until Sections 3 and 4.

2.1 Why CURE Was Developed

As noted above, the SEI has participated in numerous ITAs and “red teams.” In many of these, we identified some element related to COTS software as a major factor in failing programs. Two things soon became evident from these experiences. First, the paradigm shift to the use of COTS software is not a simple step, but instead is one that imposes a very great number of changes on an organization. Second, most organizations that have adopted a COTS approach are not aware of these changes, or have not been willing to recognize how pervasive those changes really are. As a result, a large number of organizations are falling into the same pitfalls over and over: unreasonable expectations about COTS software, misguided attempts to modify COTS products, insistence on traditional development methods when new approaches were mandatory, and other comparable behaviors inconsistent with use of COTS software.

As a result of these experiences, it was a logical step to conjecture that instead of a “red team” when a program is in jeopardy, an earlier intervention, when a program was relatively healthy and had not yet fallen into various COTS pitfalls, would be highly beneficial. Thus, the questions asked during an ITA (e.g., “Why didn’t anyone ask the COTS vendor about X?”) would be asked at a far earlier stage of a program (“Do you plan to ask the vendor about X?”). Such an assessment would actually take the form of a risk identification exercise, and the resulting method became known as the COTS Usage Risk Evaluation, or CURE.

Although it has some kinship with the risk identification method used in the SEI’s Software Risk Evaluation (SRE) technique [Higuera 96], CURE is distinct from that approach in a number of ways. For one thing, CURE focuses solely on risks relating to COTS products: it does not seek to identify the full range of risks that a program faces. Second, CURE is based on the assumption that most of these risks (i.e., the COTS-related risks) can be predicted in advance, based on evidence from the ITAs, where a small number of problems reappeared in program after program.

2.2 High-Level Description of CURE

The COTS Usage Risk Evaluation is a "front end" analysis tool that predicts the areas where the use of COTS products will have the greatest impact on the program. This allows managers to map out a strategy to address the specific risks uncovered by the evaluation, and to monitor their mitigation. CURE is designed to find risks relating to the use of COTS products and report those risks back to the organization that is evaluated. Ideally, CURE is performed on both the acquiring and the contracting organization, but this is not absolutely necessary.

The evaluation consists of four activities:

- preliminary data gathering
- on-site interview
- analysis of data
- presentation of results

For the first step of CURE, the client is asked to respond to a short questionnaire. In some cases, existing high-level briefings can provide the core of the requested information. Next, the evaluation team, usually numbering three trained evaluators, makes an on-site visit to interview the key personnel of the project. As will be described later, it is absolutely necessary that the persons interviewed truly are the decision makers: the program manager (of the acquiring side), or the project manager and chief architect (of the contractor side). This interview provides the critical data for the evaluation, and lasts an entire day. The interview makes use of an extensive set of discussion topics, which are described in Section 3.

Following the on-site interview, the evaluation team considers the data gathered using a structured analysis process. Based on this analysis, the team identifies the risks that the program faces. The analysis also yields important information about the strengths of the program. The outcome of the evaluation is presented to the client organization in the form of a briefing, delivered approximately four working days after the interview. The briefing provides a detailed analysis of risks as well as program strengths that have been uncovered by the evaluation team. This briefing is confidential, and is never shared with anyone except those who participated in the evaluation.

2.3 Audience for CURE

CURE is intended for use with any program office and contractor that are creating large-scale software systems that rely on COTS products. While the majority of its applications have been done on government programs, there is nothing inherent in CURE that makes it government specific; CURE can be used by any organization that is building large, COTS-based systems.

3 Techniques and Mechanisms of CURE

In this section we provide details on the main materials and mechanisms of CURE. These fall into two classes: those that are aimed at eliciting data, and those that help the evaluators to analyze that data. The former consist of two documents: a short questionnaire and a "discussion topics" document. The analytic materials consist of an extensive database of risk factors and conditions, as well as the detailed data that is recorded by the evaluation team.

3.1 The CURE Questionnaire

The first mechanism for data elicitation is a short document called the CURE questionnaire. This document is the initial element of CURE that is visible to the subjects of an evaluation. It is a brief, two-page document that is sent to the client organization several weeks in advance of the on-site interview, to be filled in with short answers and returned to the evaluation team. The purpose of the questionnaire is to provide to the evaluators some advance insight into the project: its overall purpose, size, and scope, the nature of the system to be built, the identities of the key players, and so forth. It is not intended to provide detailed, exhaustive information, but only to permit the evaluators to become acquainted with the program's context, its terminology and acronyms, and so forth, before the on-site interview.

When the questionnaire is returned to the evaluation team, it provides the basis for the initial inbrief (described more fully in Section 4.3). The questionnaire also permits the evaluation team to determine guidelines for focusing the discussion topics on the key issues relevant to the program at hand. After this, the questionnaire plays no further role in the CURE process. The following excerpt shows 2 (of 19) items from the questionnaire.

1. Mission need for the system

- **Briefly describe the mission to be fulfilled by the system, and characterize it as: replacement, update, modernization, creating a new system, other.**
- **If replacement, update, or modernization is involved, briefly describe the differences between the existing system and the planned system.**
- **Indicate the planned overall lifetime of the new system.**

2. Business/technical processes

- **Briefly describe the business or technical process that the new system will support.**

3.2 The Discussion Document

The conceptual basis of CURE lies in an extended document that enumerates a number of COTS-related *discussion topics.* These topics include such things as COTS-related requirements, design, and testing; they also include such things as the experience base of a project's main decision makers, contractual relationships between and among COTS vendors, and plans for COTS product sustainment and maintenance. Taken together, these discussion topics provide the foundation for CURE: they guide the on-site interview, provide a focus for the evaluators' analysis, and are critical to understanding the COTS-related risks that emerge as the evaluation's output. The discussion document is divided into the following fifteen chapters:

- Part One: General
    - 1 System Description
    - 2 Use and characteristics of COTS products
    - 3 Management readiness
    - 4 Technical readiness
- Part Two: Business
    - 5 Budget, schedule, and contracts
    - 6 Vendors and suppliers
- Part Three: Infrastructure
    - 7 Standards
    - 8 Process
- Part Four: Lifecycle
    - 9 Evaluation
    - 10 Requirements
    - 11 System design
    - 12 System integration
    - 13 System testing
    - 14 Fielding
    - 15 Maintenance and sustainment

Each of these chapters is roughly a page in length, and is organized around a small number of general areas which in turn are broken down into individual discussion topics. The topics cover many different aspects of a program, but are primarily concerned with COTS-specific issues. Figure 1 illustrates Chapter 5 ("Budget, Schedule, and Contracts"). This example shows six general areas, each of which contains two to four discussion topics. (Each item, whether bulleted or sub-bulleted, is considered a separate discussion topic.)

5.1 Contractual relationships

- **Contractual relationships with and between all entities that will contribute to the project**
  - **Relationships between different organizations**
  - **Relationships between different components of the same organizations**
- **Nature of any other contributions (i.e., non-contractual contributions) that are expected**

5.2 Contract flexibility

- **{For any contract that exists:} Provisions in the contract that provide flexibility (e.g., with respect to changed costs, extensions of vendor services, increased requirements, etc.).**
- **Previous contracts with similar COTS-related provisions that served as a contract model**

5.3 Contract renegotiation

- **{For any contract that exists:} Contract renegotiation that has occurred that was driven by COTS-related issues**
  - **Effects of this renegotiation**

5.4 Cost projections and budgeting

- **Methods used for cost projections and budgeting for the system; impact of COTS products on these methods**
- **Resource allocations for specific COTS-related activities (e.g., standards group participation, technology watch)**

5.5 Product commercialization

- **Contractual provisions concerning commercialization of modified products**
- **Expectations regarding other customers of the modified products**

5.6 Project schedule

- **Schedule for the system in terms of major milestones, IOC, interim deliverables, major decision points**
- **Provisions in schedule for COTS products releases**
- **Any other COTS-related factors that contribute to or constrain the schedule**

Figure 1: Chapter 5 of the Discussion Document

Note that the discussion topics are not themselves statements of risk, and the mode of expression in which each topic is phrased is significant. Each is intended, insofar as is possible, to be entirely aimed at elicitation of facts, and contains no implicit judgment about any risky condition. Thus, in the first topic from the above example

- **Contractual relationships with and between all entities that will contribute to the project** the topic is a neutral basis for a discussion of whatever contractual relationships might exist for a given project. The presence or absence of risk will be determined later, after a full analysis of all data has occurred.

The reader will find, in examining the discussion topics document as a whole, that a certain redundancy exists between the different chapters of the document. This redundancy is unavoidable, and even necessary. For instance, some of the points covered in Chapter 2 (“Use and Characteristics of COTS Products”) are also revisited in Chapter 9 (“Evaluation”), Chapter 10 (“Requirements”), and so forth. The need for this overlap is that there are subtle, although quite real, distinctions in the kinds of data that will emerge from discussing COTS products from these various perspectives.

The discussion document is provided to the personnel of the program several days in advance of the on-site visit, and serves as the outline and agenda for the interview.

3.3 The Analytic Materials

The analytic materials used in CURE consist of the data that is gathered by the evaluation team, and a database relating risk factors to risk conditions.

During the on-site interview, the participants led by the evaluation team step through the discussion topics in order. As they do so, the evaluators seek to elicit sufficient data to permit an adequate analysis. The precise data that are of interest will depend on the nature of the project that is being evaluated. For instance, one project may be developing a very long-lived system for which ongoing renewal of product licenses is a potentially critical issue. In such a case, for the following discussion topic

- **Estimated costs for future license renewal; basis for this estimate**

the evaluation team might be interested in the following kind of data:

- Are the current estimates based on discussions with vendors?
- Is there any anticipated change in license structure?
- Is there any current knowledge about future cost increases? If so, what is the source of that knowledge?

The granularity of data such as this is quite detailed and low-level. On the other hand, if the system to be built does not have a long-term focus, the question of future license costs may be much less important, and this particular discussion topic might be covered only briefly.

After the data on all relevant topics has been gathered, the CURE method then makes use of a database that contains two kinds of items, *risk factors* and *risk conditions,* both of which exist initially in template form. Risk factors are neutral statements of fact that provide a focus for the data-gathering activity. These factors are instantiated and made specific as the evaluators record the data and perform their analysis of it. The goal is to summarize the raw data into succinct descriptions of some particular aspects of the project or system under evaluation. For instance, for a generic risk factor such as

> <*system*> has <*some degree of dependence*> on <*other system*>

the data might indicate that it should be instantiated as

> The ABC system has total dependence for all of its data storage on the XYZ system which is also currently in development.

The other contents of the database, risk conditions, represent the first explicit mention of risk, in the sense of a potentially harmful state that the program might face. Like the risk factors, the risk conditions are initially templates, and are phrased as general statements relating to COTS-specific risks:

> The quality of product support for <*product x*> by <*party*> will be <*some deficiency*>.

As the analysis proceeds, these generic templates are instantiated based on the risk factors, which are in turn based on the data from the interview. For instance, the instantiation of the risk condition above, supported by its (hypothetical) related risk factors, might be that

The quality of product support for SomeBudgetTool by QED Software will be unpredictable and erratic, because

- There exists no evidence that SomeBudgetTool has been successfully used in other systems.
- User groups for SomeBudgetTool do not exist.
- The entity with responsibility to support SomeBudgetTool is QED Software.
- The product support process consists of QED’s 24/7 help desk which exists right now, but it is not likely that it will continue beyond the end of the year.

4 Conducting a CURE

In this section we describe in detail how a CURE should be conducted. We describe the CURE interview process, and also some other activities of the evaluation team during the on-site visit.

4.1 Timing

The optimal time for CURE is when it can have the greatest impact on program direction. Thus, when applied early in the program, the evaluation can help in planning for important decisions ahead. The more major COTS-related decisions have already been made, the more the potential impact of CURE declines. By the point that all decisions have been made, when all COTS products have been selected, licenses purchased, and so forth, there is little value in conducting a CURE for a program.

4.2 The Questionnaire

The questionnaire is sent to the client organization four weeks in advance of the on-site visit. The organization is instructed that the purpose is merely to acquaint the evaluation team with the general outlines of the project, and not to elicit detailed data. Hence, the answers are expected to be concise, normally one or two paragraphs. The client organization is also warned against submitting extensive documentation about the program; whatever low-level data about the program is needed will be brought out during the on-site interview.

4.3 The CURE Interview

We have already noted that the interview must be with the key decision makers of a project: the program manager, the chief architect, and so forth. The reason this is mandatory is that the type of data that is sought by the evaluation is only available from those persons. For instance: who is making decision X? What are the factors that will influence that decision? What other factors have been considered?

The CURE interview will generally require between 7 and 10 hours to conduct. It is recommended that the participants complete the interview in a single day, although this may not always be possible. The interview is conducted by considering each area and each discussion topic in turn. The evaluators should avoid a simple question-and-answer approach and engage the participants in a give-and-take discussion of each topic.

The very first activity of the interview is an inbrief lasting roughly thirty minutes, and delivered by the CURE team. The inbrief has two sections. The first is a description of the CURE process describing what will occur both on the interview day and during the succeeding four days of analysis by the evaluation team. The second part of the inbrief is a description of the program as the evaluation team understands it. The value of this is twofold. First, it clarifies for the decision makers how well the evaluators do (and do not) understand their program. Second, and more important, experience has shown that at the start of the CURE interview, the participants have a tendency to try to get as much information on the table as possible. While laudable, this is usually a problem for the evaluators. Topics are raised prematurely, and too much information is brought into the discussion too quickly. When the appropriate occasion for those topics arises, there is an unavoidable sense of redundancy. The inbrief is a useful way to correct this problem and enter into the discussion of the program in a measured and methodical manner.

The major part of the interview is a step-by-step progression through the discussion topics. These topics have been formulated to cover the principal COTS-related issues that most programs will face. However, not all discussion topics are relevant for every evaluation. For instance, if the issue of commercialization of modified COTS products has no relevance for the program at hand, then Section 5.5 of the discussion topics document may be omitted. The evaluation team will make decisions on which topics are relevant as the interview progresses.

The printed order of discussion topics will likely be followed in only a general manner. Thus, the precise order in which each set of bulleted and sub-bulleted items is addressed may vary: discussion might proceed from a general point to a more specific one, or may immediately deal with specifics. For instance, if a particular topic area contains three bulleted items and two sub-bulleted topics, the discussion may begin with any of these, in any order, as the evaluators deem appropriate for the program at hand. There is no need to proceed exactly according to the printed order within a single area, so long as all relevant items are covered in some manner.

4.4 Data Analysis

During the three or four days following the interview, the evaluation team performs a structured analysis of the data gathered. Using the database of risk factors and risk conditions described in Section 3, the team formulates a set of COTS-related risks that appear to be significant for the program. These risks are then ranked in order of criticality, severity, or imminence to the program.

In addition, the analysis produces a complementary set of program strengths. That is, if the data and analysis indicate that the program is aware of some COTS-related risk, and has taken steps to mitigate that risk, then that issue is regarded as one for which the program's management is to be complimented.

Although there is no set number of either risks or strengths, the evaluations to date have tended to identify 8 to 12 of each.

4.5 The Outbrief

The result of the analysis is recorded in an outbrief that is delivered to the program no later than four days after the interview. The outbrief lists the risks and strengths, and for each risk, the evaluation team proposes one or more mitigations to that risk. These mitigations are based on experiences with other programs, other CUREs, and ITAs that have identified the same or a similar risk.

As noted earlier, the outbrief, and in fact the entire CURE process, is confidential. The results of the evaluation are given to the participants and to no one else. It is entirely the decision of the participants whether the results are shared with any other persons.

5 Results to Date

In this section we describe the kinds of evaluations we have carried out, and also summarize the general character of the risks that we have identified in these evaluations.

5.1 Kind of Organizations and Programs

To date, we have performed CUREs on the DoD, other government agencies, and industrial organizations. The projects have included weapons systems, information systems, and command, control, communication, and intelligence (C3I) systems. Some of the evaluations have been on only one of the parties to an acquisition (i.e., only the acquirer or the contractor) while others have included both sides. The results indicate that the process is working as we had hoped. Most programs have expressed considerable interest in the process, and the outcomes of most evaluations have been welcomed by the organizations involved.

One problem that we have noted, and for which there appears no ready solution, is the intrusive nature of CURE. The evaluation requires one full day of dedicated participation by the key personnel of a project, and this is seldom a welcome prospect for any organization. A Program Manager or a Chief Architect is generally an extremely busy person, and demanding that someone in that role give up an entire 10-hour day is placing a great burden on the program. We are grateful that, in most cases, the personnel involved have realized that if CURE is to have genuine value, it requires that the key decision makers participate in the on-site interview.

One major lesson learned is that for organizations and projects that are in relatively good condition, few of the risks we identify are surprises to the participants. For such projects, the results of CURE tend to be an external affirmation of good management practice. (It should be noted that on most occasions, even for well-managed projects, at least some of our findings were unexpected to the organization’s personnel.) However, for organizations and projects that are in a less healthy state, our findings typically point out risks that are unforeseen by the program’s participants. On the one occasion where we identified a large number of extreme and severe risks, we predicted that the risks to the program were probably beyond mitigation; on that occasion, the program was cancelled within a year.

5.2 Statistical Breakdown of Results

We now list the major areas of risk that we have identified during the past four years. Note that we are here summarizing only the *areas* where we have identified risks. We do not provide examples of actual risk statements, since we always guarantee that the result of a CURE evaluation is completely confidential.

To make this summary, we rephrased and generalized the results of all of the CUREs to date. For instance, we had evaluated several programs in which we found risks associated with the program's use of integrated product teams (IPTs). Restated to remove any connection with a specific program, they included

- the expected composition of IPTs is insufficient for this program because it is top-heavy with people from <*only one part of the program*>
- the expected composition of IPTs is insufficient for this program due to the absence of any empowered persons on the IPTs
- the management team lacks any experience with using IPTs at all

and other comparable statements of a risk condition. We then generalized all of these into a higher level risk area relating to weakness of the program's IPTs, due to inexperience, poor structuring, or other such factors. We also calculated, for each risk area, its overall percentage in the programs we have evaluated.

The result is a collection of 21 risk areas that fall into 4 broad categories: Programmatics and Management, Technical Areas (e.g., Design, Evaluation, Testing), Mission and Stakeholders, and Product-Specific Issues. It is important to note that some of the risks are not truly related to COTS software usage. Thus, though most risks have some kind of COTS product focus (e.g., the lack of sufficient COTS-related experience on the part of the management team), some were quite independent of COTS concerns (e.g., harmful personnel shifts within the management team, or unsatisfactory use of IPTs as cited above). We have found it impossible to avoid noting such major risks, regardless of their pertinence to COTS issues, and we saw no need to suppress these "non-COTS" risks from this tabulation.

The proportion of risks *across* the categories was somewhat surprising. We had expected that risks indicating difficulties with particular COTS products themselves would be most common, but this was not the case. The largest category consisted of risks in the category of Programmatics and Management. It is possible that because CURE has a bias toward management issues, a preponderance of risks will occur in that category in all cases. This will be proven or not as we get more data.

Of the individual risk areas, "Fulfilling System Requirements" was the single area holding the largest amount of risk by far (12%), which was not unexpected. Note also that since we distinguished "Requirements Management" from "Fulfilling System Requirements," we may have even understated the importance of requirements; counted together, these two represent some 16% of all of the risks.

This is the breakdown of risks by category[3]:

Programmatics and Management:	31%
Technical Areas:	24%
Mission and Stakeholders:	30%
Product-Specific Issues:	16%

Table 1 shows the individual risk areas within each category.

Programmatics and Management	Personnel (7%)
	Budget (5%)
	Requirements management (4%)
	Contract (4%)
	Risk management (3%)
	Decision making (3%)
	IPTs (3%)
	Process (2%)
Technical	Business processes (5%)
	Configuration management (5%)
	Fielding (4%)
	Test (4%)
	Integration (3%)
	Data conversion (2%)
	Evaluation (1%)
Mission and Stakeholders	Fulfilling system requirements (12%)
	Sustainment (9%)
	Stakeholders (9%)
Product-Specific Issues	COTS modification (6%)
	COTS version skew (5%)
	COTS product support (5%)

Table 1: Risk Categories and Risk Areas

[3] Note that these percentages add up to 101% because of rounding.

6 Summary and Future Work

The CURE method has proven to be a useful tool for organizations that acquire or develop COTS-based systems. It provides those organizations with useful insights about COTS-based development that are derived from a large number of other programs, and crystallizes the experience gained from several years of observation of software system acquisitions, both successful and otherwise. However, there are several improvements that can be made to the existing method. They are addition of some additional topic areas; automation of portions of the analysis process; and finalization of educational materials about the method.

The discussion topics presently cover most, but not all pertinent areas where COTS-related risks can affect a program. Of the ones missing, by far the most significant one is that of security. While some discussion topics touch on security, we feel that it would be desirable to introduce an entirely new chapter to the document where several issues relating to security and COTS products could be considered systematically.

The analysis process would benefit from additional computer support. Currently, the evaluators do a considerable amount of the analytic work by hand, and it would be useful if a good deal of this were automated. This may also contribute some greater efficiency to the process; if the number of days of analysis were reduced by one, the cost of CURE would be lowered.

Finally, the process by which new CURE evaluators are trained is largely one where experienced CURE evaluators mentor new evaluators. A training manual is currently in preparation, but it would also be desirable if course materials were to be developed as well.

As these materials are developed, we believe that CURE could become a widely-used mechanism, and one that is readily available to organizations and enterprises that wish to improve the manner in which they acquire and develop COTS-based systems. It is our hope that organizations other than the SEI will become familiar with the CURE method, and will adopt it as a useful element in their pursuit of improved software engineering processes.

References

[Albert 02] Albert, C. & Brownsword, L. *Evolutionary Process for Integrating COTS-Based Systems (EPIC): An Overview* (CMU/SEI-2002-TR-009, ADA405844). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. <http://www.sei.cmu.edu/publications/documents/02.reports/02tr009.html>.

[Baker 02] Baker, T. “Lessons Learned Integrating COTS into Systems,” 21-30. *Proceedings of the First International Conference on COTS-Based Software Systems (Lecture Notes in Computer Science, 2255)*. Orlando, FL, February 4-6, 2002. Berlin, Germany: Springer-Verlag, 2002.

[Dorofee 96] Dorofee, A., Walker, J., Alberts, C., Higuera, R., Murphy, R. & Williams, R. *Continuous Risk Management Guidebook*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.

[Hayes 02] Hayes, M. “Hershey’s Biggest Treat: No Tricks.” *InformationWeek*. Oct. 29, 2002. <http://www.informationweek.com/story/IWK20021029S0005>.

[Higuera 96] Higuera, R. & Haimes, Y. *Software Risk Management* (CMU/SEI-96-TR-012, ADA315789). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996. <http://www.sei.cmu.edu/publications/documents/96.reports/96.tr.012.html>.

[Pfarr 02] Pfarr, T. & Reis J.E. “The Integration of COTS/GOTS Within NASA’s HST Command and Control System,” 209-221. *Proceedings of the First International Conference on COTS-Based Software Systems (Lecture Notes in Computer Science, 2255).* Orlando, FL, February 4-6, 2002. Berlin, Germany: Springer-Verlag, 2002.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY
(Leave Blank)

2. REPORT DATE
September 2003

3. REPORT TYPE AND DATES COVERED
Final

4. TITLE AND SUBTITLE
Identifying Commercial Off-the-Shelf (COTS) Product Risks: The COTS Usage Risk Evaluation

5. FUNDING NUMBERS
F19628-00-C-0003

6. AUTHOR(S)
David J. Carney, Edwin J. Morris, Patrick R. H. Place

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER
CMU/SEI-2003-TR-023

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
HQ ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2116

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESC-TR-2003-023

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified/Unlimited, DTIC, NTIS

12B DISTRIBUTION CODE

13. ABSTRACT (MAXIMUM 200 WORDS)

The expansion in use of commercial off-the-shelf (COTS) products has been accompanied by an increase in program failures. Many of these failures have been due to a lack of familiarity with the changed approach that COTS products demand. This report describes the development of an approach to reduce the number of program failures attributable to COTS software: the COTS Usage Risk Evaluation (CURE). The origin of CURE and an overview of the method, along with detail on the materials and mechanisms used in CURE, are provided. The CURE process is outlined and the results of the evaluations that have been conducted are summarized. Finally, possible future directions for CURE are explored.

14. SUBJECT TERMS
COTS, risk, evaluation, commercial software

15. NUMBER OF PAGES
34

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
Unclassified

20. LIMITATION OF ABSTRACT
UL