Carnegie Mellon
Software Engineering Institute
Pittsburgh, PA 15213-3890

# Guidelines for Using OAR Concepts in a DoD Product Line Acquisition Context

John Bergey
Dennis Smith

*March 2000*

**Product Line Practice Initiative**

**Technical Note**
CMU/SEI-2000-TN-004

## About the Technical Note Series on Reengineering Practices for Product Lines

The Product Line Systems Program is publishing a series of technical notes designed to condense knowledge about the use of reengineering practices for the Department of Defense (DoD) acquisition manager and practitioner. Each note will focus on one aspect of applying reengineering practices to adopting software product line practices in the DoD. These notes provide practical guidance to early adopters on ways to integrate sound reengineering practices into their product line acquisitions. By investigating best commercial and government practices, we hope to address current challenges and increase the understanding, maturation, and transition of this technology. This current note focuses on providing guidance for DoD organizations for mining legacy systems to obtain core assets that will fit into a software architecture for a product line. Future technical notes will expand on other acquisition examples and provide additional guidelines for using Options Analysis for Reengineering (OAR) in DoD product line acquisitions.

This series is a companion to the SEI series on product line acquisition and business practices. Together, these two series of technical notes will lay down a conceptual foundation for DoD reengineering and product line business and acquisition practices that is consistent with the SEI's Product Line Practice Framework [Clements 99]. Other information is available on the SEI's Web page at http://www/sei.cmu.edu.

# Contents


Acknowledgements iii

Abstract v

1 Introduction 1

2 Product Line Context for Reengineering: Mining and Adapting Legacy System Assets to Obtain Core Assets 3

3 The Relevance of OAR to Mining Legacy Assets for a DoD Product Line 5

4 Application of OAR to a Specific DoD Acquisition Example 7

5 Summary and Conclusions 13

References 14

Feedback and Contact 16

## List of Figures


Figure 1: The Horseshoe Model Underlying OAR 6

Figure 2: Contractual Process for DoD and Government Acquisitions 8

## Acknowledgements

We wish to acknowledge the insightful comments made to earlier versions of this document that substantially strengthened it. These comments were made by

- Carl R. Ruzicka, Contracting Officer, Naval Air Warfare Center, Aircraft Division, Lakehurst, N.J.

- Liam O'Brien, Member of Technical Staff, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa.

# Abstract

Many Department of Defense (DoD) organizations are considering product line initiatives as a means of overcoming the issues of quality, cost, and schedule inherent in a “one-at-a-time” system development or acquisition paradigm. Because a product line approach revolves around the creation of a comprehensive set of core assets, mining and adapting (i.e., reengineering) existing legacy system assets can offer significant leverage. By mining and adapting these assets, an organization can exploit the proven capabilities of existing systems, reduce the funding required, and develop/acquire new systems of higher quality within a shorter time frame.

This technical note focuses on providing guidance for DoD organizations for mining legacy systems to obtain core assets that will fit into a previously defined software architecture for a product line. We explain how insights from a conceptual model, Options Analysis for Reengineering (OAR), can be used in an acquisition context to provide the government with an approach for obtaining greater insight and understanding into a contractor’s proposed technical reengineering approach.

# 1 Introduction

Software is pervasive in modern defense systems. Software systems developed around product lines offer the Department of Defense (DoD) the opportunity for leveraging the commonality between systems. Such an approach, in sharp contrast to the "one-at-a-time" system development/acquisition paradigm, enables substantial savings in time and schedule, as well as better quality.

The SEI's Product Line Practice Framework [Clements 99] details the practice areas that need to be addressed for the successful development or acquisition of a software product line. One of these practice areas, "mining existing assets," represents a crucial starting point for a product line effort. By mining and suitably adapting legacy system assets, an organization can often exploit the proven capabilities of their existing systems, reduce the initial funding that is needed, and shorten the time required to achieve an initial product line capability.

Traditional reengineering techniques usually focus at the code level, and as a result do not provide leverage for updating to modern software architectures. In addition, they are time-consuming and costly. However, recent architecture reconstruction techniques [Kazman 97] provide a basis for extracting the as-built architecture from the existing code, and using this information as a foundation for further evolution.

In conjunction with the architecture reconstruction work, a conceptual "horseshoe" model has been developed to distinguish different levels of reengineering analysis and provide a foundation for transformations at different levels, especially for transformations at the architectural level [Carriere 99]. This model outlines a rich set of technical choices that reengineers make in adapting software systems. However, because of its technical focus, it has not been accessible to decision makers in a form that can assist them in deciding on complex options regarding the future of their legacy systems.

A previous technical note [Bergey 99a] introduced Options Analysis for Reengineering (OAR), a conceptual approach for analyzing reengineering options to enable better software reengineering decision making. In the current note we address how concepts derived from the OAR approach can be used by the DoD acquisition organization for making informed decisions about mining legacy assets as components for a product line architecture.

In this technical note, we first provide background on legacy assets in product line systems, followed by a discussion of how OAR concepts relate to the mining of assets for a product line. Next, we provide a high level overview of the DoD acquisition process. With this

background established, we provide acquisition guidelines for leveraging legacy software assets as product line core assets.

Because this is the first version of a planned series of technical notes, the reader is advised to consult the SEI Web site (www.sei.cmu.edu) to determine the availability of related and follow-on notes and reports. Future notes in this series will provide further elaboration and examples.

# 2 Product Line Context for Reengineering: Mining and Adapting Legacy System Assets to Obtain Core Assets

## 2.1 What Role Does Reengineering Play in a Product Line Approach?

A software product line is a set of software-intensive systems sharing a managed set of features that address a particular market segment or fulfill a particular mission. Substantial economies are achieved when the systems are developed from a common set of organizational assets. A product line approach enables the systematic leveraging and reuse of a common set of software core assets.

In most situations, architects making plans for a product line are faced with legacy systems that have been developed over many years at a substantial cost. These legacy systems represent a patchwork of mainframe, minicomputer, and desktop applications, both centralized and distributed, under dispersed control. They can be fragmented by programming language, geography, database incompatibilities, operating system, and corporate mergers. Nevertheless, there is a requirement to maximize the information system assets by protecting, managing, integrating, and modernizing the legacy systems. In order to leverage models, architectures, designs, documentation, testing artifacts, people, processes, and implementations, product line planning focuses on strategic, coarse-grained reuse.

## 2.2 What Types of Concerns Need to be Addressed in Mining Assets?

Reengineering is complex. It requires carefully analyzing candidate reengineering options and strategies that affect the interests of many stakeholders, and involves making non-trivial tradeoffs about technical, programmatic, and organizational considerations. In general, reengineering decision-making requires

- considering a diversity of (reengineering) options, each of which may involve significant tradeoffs
- performing analyses to compensate for a lack of up-to-date legacy system requirements/design documentation
- exploring and resolving the uncertainties about the implementation of a legacy system including its functionality, integrity, and quality attributes
- obtaining extensive quantitative and qualitative data on which to base decisions
- exploring the impact of reengineering from the perspective of multiple stakeholders and resolving conflicts stemming from a fracturing of software expertise
- integrating technical and programmatic constraints and coalescing decision making from a unifying perspective

## 2.3 How Does Mining of Assets for a Product Line Differ from Traditional Reengineering?

Traditional reengineering is challenging because it often attempts to evolve a legacy system when the system itself has reached a point where maintenance and enhancement practices are no longer adequate. Moreover, since software life cycle maintenance activities typically involve corrective, perfective, and adaptive maintenance and “white box” software enhancements, code level considerations are often a focal point.

In mining legacy assets for a product line, the primary focus is not on either code level transformations or reengineering a system in its entirety. Rather, the focus is on evaluating how specific legacy software assets can be mined and adapted for product line usage (e.g., as start-up core assets). As a result, the emphasis is on architectural compatibility and interface considerations and involves isolating large-grained “chunks” of legacy system functionality to “black box” software elements so they can be suitably adapted or wrapped to serve as core assets. Thus, there is a fundamental shift from code level scrutiny (e.g., white box considerations) to focusing on isolating functionality along “black box” lines and on architectural extraction/reconstruction. Key tradeoffs must be made in the early exploratory phases of mining assets for product lines. The technical challenges center on evaluating which individual legacy software assets can best be mined and adapted for product line usage as core assets. In some cases, mining of legacy assets may only be a stop gap measure to quickly obtain a start-up set of core assets. The assets may not be suitable for the long term, since they were not originally designed to accommodate the commonality and variability of a family of similar applications. In other cases, it may be determined that the mining of assets, although superficially attractive, is not really practical or cost effective.

# 3 The Relevance of OAR Concepts to Mining Legacy Assets for a DoD Product Line

OAR is a conceptual approach to analyzing and understanding reengineering options. While the OAR approach is still evolving, the concepts can be applied in DoD and government acquisitions to obtain added insight and a more comprehensive understanding of a contractor's proposed approach for reengineering legacy system assets. To use the insights from OAR effectively in the DoD, it is necessary to understand its technical underpinnings (i.e., how it applies to any reengineering effort) and have a suitable set of acquisition guidelines.

The OAR approach, when fully evolved, will provide a unified approach for analyzing reengineering options and evaluating candidate reengineering strategies. OAR's underlying model will

- combine reengineering and architectural views of software analysis and evolution with defined mappings between these views
- classify and stratify reengineering analysis and implementation options/approaches into distinct layers with explicit mapping between layers
- provide key information for making informed choices about the appropriate time and circumstances in which to use each of the reengineering options/approaches
- codify technical and non-technical issues and risks for each of the reengineering options/approaches
- relate organizational and programmatic factors to *reengineering option* decision making in a system and product line context

OAR's "horseshoe" model (Figure 1), which combines reengineering and architectural views of software analysis and evolution, is briefly outlined below to demonstrate how it can apply to DoD acquisition issues. For more details on OAR, see [Bergey 99a].

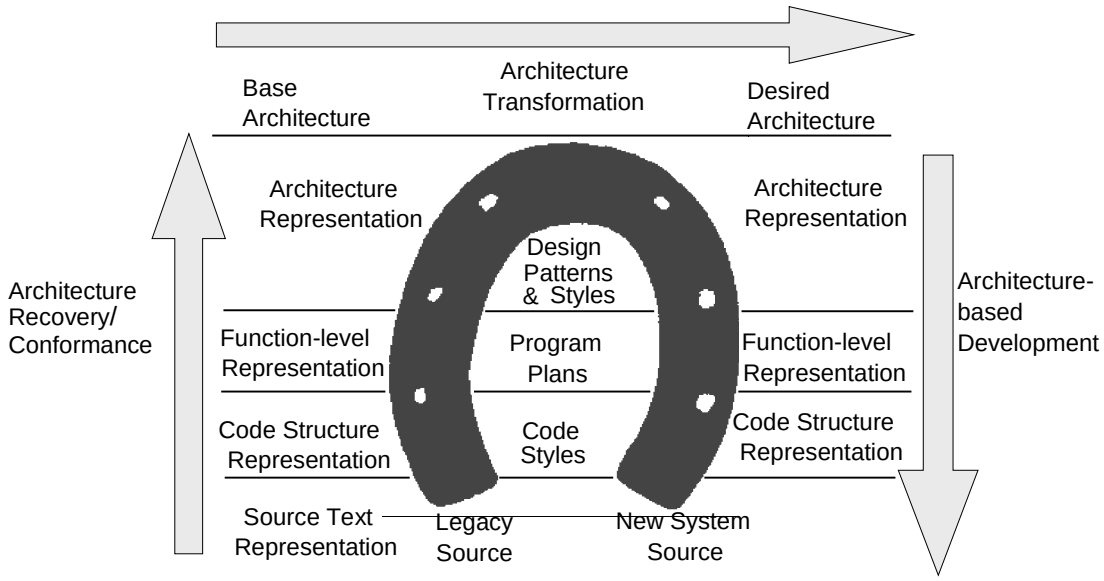


*Figure 1: The Horseshoe Model Underlying OAR*

### 3.1 OAR's Horseshoe Model

The purpose of the visual metaphor is to integrate the code-level and the architectural reengineering views of the world. The three basic reengineering processes form the basis of the horseshoe: analysis of an existing system, logical transformation, and development of a new system.

The first process goes up the left leg of the horseshoe. It starts at the source code level and aims to recover the architecture by extracting artifacts from the code and its associated abstract syntax tree.

The second process, architectural transformation, goes across the top of the horseshoe. In this case, the as-built architecture is recovered and then reengineered to become a desirable new architecture. It is re-evaluated against the system's quality goals and subject to other programmatic and economic constraints.

The third process, which goes down the right side of the horseshoe, uses Architecture-Based Development (ABD) [Bass 99] to instantiate the desired architecture. In this process, packaging issues are decided and interconnection strategies are chosen. Code-level artifacts from the legacy system are often wrapped or rewritten to fit into this new architecture.

Within a DoD acquisition environment, there will be a need to initially obtain visibility into the decisions and processes used to recover and transform assets at the left side and top of the horseshoe. In addition, the integration of assets into the product line architecture at the right side of the horseshoe will guide the crucial task of fitting existing coarse-grained components into a product line architecture.

# 4 Application of OAR Concepts to a Specific DoD Acquisition Example

It is critical to have a systematic approach to understanding the viability of a contractor's proposed approach as early as possible in the acquisition process. Such insight can be used as a proactive means of

- providing early visibility into the transformations and critical design decisions that will drive the core asset development effort
- obtaining insight into how compatibility of the reengineered assets with the selected product line architecture will be achieved

To aid in this understanding, we will apply the insights of OAR to a DoD acquisition with the following characteristics:

- A product line architecture has been developed (or selected) and documentation is a GFI.
- The relevant acquisition requires the exploration and mining of legacy assets to obtain a set of core assets for the product line.
- The contractor is to use insights from OAR's horseshoe model to guide them in obtaining the necessary system level understanding of the legacy system and its assets.
- The contractor is to identify the key issues and tradeoffs, establish the practicality of mining assets, identify the necessary transformations, cost and schedule, and risks.
- The DoD acquiring organization will incorporate elements of OAR's horseshoe model in its technical evaluation criteria to create a "level playing field" among contractors and assist in the source selection process.

The typical DoD acquisition process is outlined next in Section 4.1. Key acquisition issues that relate to mining and adapting legacy system assets for a product line are then highlighted in Section 4.2. Building on these two subsections, Sections 4.3 and 4.4 provide specific guidelines derived from OAR concepts that apply to designated aspects of the acquisition process.

## 4.1 Outline of the DoD Acquisition Process

The three primary phases of the acquisition process, the pre-award, award, and post-award phases are illustrated in Figure 2.

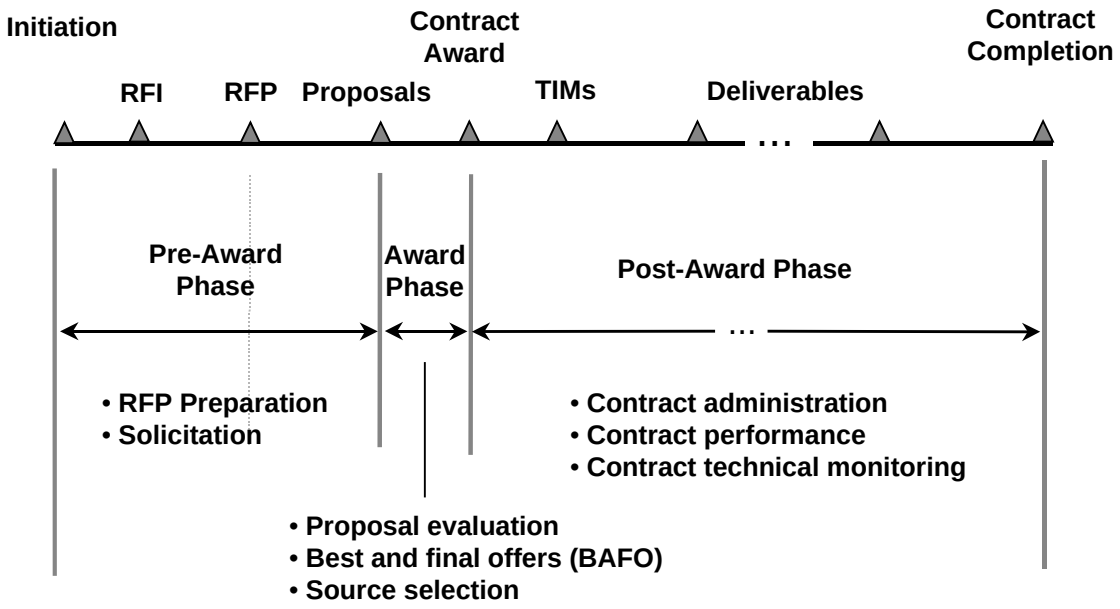


*Figure 2: Contractual Process for DoD and Government Acquisitions*

The request for proposals (RFP), which is the primary outcome of the pre-award phase, includes a statement of work (SOW), performance and delivery specifications, preparation instructions to offerors (on preparing proposals), and the technical evaluation criteria that will guide the source selection process.

The award phase can include requirements for technical presentations, "proof-of-concept" demonstrations, or even "bid samples" for evaluation by the contracting organization. Such requirements are often added as risk mitigation measures that can directly influence contract award to the extent that they are part of the acquiring organization's source selection plan and technical evaluation criteria.

During the post-award phase the contractor is responsible for performing the work described in the SOW and delivering the specified products and services. The contractual requirements that define these products and services are usually expressed in terms of quality attributes (e.g., functionality, performance, security, interoperability, modifiability), quantity, and cost and schedule.

Detailed information on the federal acquisition regulations (FAR) that govern all DoD acquisitions is found in [DoD 98]. Additional background information on how software product lines specifically relate to the DoD acquisition environment can be found in [Bergey 99b].

## 4.2 Key Acquisition Issues in Establishing a Product Line

When migrating to a product line approach, two overriding concerns that relate to mining and adapting legacy system assets are the following:

1. "Can a competent software contractor other than the original developer gain a sufficient understanding of the legacy code (through exploration and analysis) to be able to successfully mine the legacy software?[1]

   [*A negative answer is a primary reason why acquiring organizations are inclined to let a sole source contract to the original developer and are reluctant to go competitive. In reality, though, the original development contractor may not be any better equipped or knowledgeable to mine the legacy code than a third party contractor.*]

2. "Can effective proactive acquisition measures be taken to detect potential problems early in the contracting cycle before a substantial amount of funds have been expended"?

   [*Even in the case of a sole source contract, an acquiring organization would like to have a contractual "safety-valve" should things go awry.*]

From the standpoint of a DoD acquisition organization, these two questions often represent the crux of the risk of mining and adapting legacy software assets—especially when considering a competitive acquisition. There are, fortunately, risk mitigation measures that can be enacted to substantially address these issues and reduce the risk of being irreversibly "trapped" into a course of action should a contractor falter in the process, or if the mining of assets becomes infeasible or impractical.

These issues focus on recovering if things go awry. However, the primary goal for using an OAR based approach is to "raise the bar" with regard to contract performance. These concepts can require the winning contractor to present a well thought out, enactable process that details the architectural implications of reengineering decision making. Hopefully, this will avoid any mismatch in acquirer/performer expectations and result in a level of performance and thoroughness that otherwise would not be achievable.

## 4.3 Guidelines for Contract Award and Monitoring

Given the standard DoD acquisition process (Figure 2), insights from OAR can be used as a risk mitigation strategy in obtaining a set of core assets for a product line. It is applicable to a competitive or sole source acquisition. The approach is described in terms of specific guidelines that prescribe the steps to be taken:

[1] Going competitive assumes that the government owns the data rights to the existing legacy software. This may require a careful investigation of what the government's *legal rights* are as far as "use" is concerned, including any provisions for buyout of rights. Even if there are no buyout provisions explicitly stated in the software license/contractual agreement(s), this may/may not be a big issue depending on whether the original contractor still has an interest (and/or staff) in business aspects of the legacy software. These matters would have to be confirmed and/or negotiated before the acquiring organization could make a determination as to whether they can legally go competitive.

1. Develop the acquisition strategy around an *indefinite-delivery/indefinite-quantity (ID/IQ)* "task order" contract to be competitively awarded to the contractor submitting the "best value" proposal.

   [*The benefit of this approach is that the government does not have to commit to give a contractor more than one task order under an ID/IQ contract.*]

2. Determine the "best value" technical evaluation based on the efficacy of the contractor's *proposed process* for exploring and analyzing the legacy software and extracting the "as built" software design at the higher levels of abstraction identified in OAR.

   [*In the Proposal Preparation Instructions that are part of the RFP, the DoD acquisition organization will include guidelines for what the offeror should include (in their proposal) with regard to their proposed process. In the case of a sole source contract, the down side is that there would obviously be only one proposal to consider.*]

3. Negotiate a first task order under the ID/IQ contract, concurrent with contract award, for a relatively small sum of money corresponding to the declared contract minimum.

   [*This approach provides the government with more leverage than would otherwise be possible and should expedite the negotiation process for the first task order.*]

4. Include a number of pivotal tasks in the first task order that will decisively show the contractor's ability (or inability) to perform critical tasks that will mitigate high risk items associated with mining the legacy assets to achieve an initial set of product line core assets. Some of these risks include the inability to understand legacy assets, the inability to make abstractions to the architectural level, or mismatch between legacy assets and product line architecture.

5. Structure each of the tasks in the first task order so that they will result in tangible deliverables that can be evaluated by a technical agent of the contracting organization (or by an objective third party) to determine their suitability (or unsuitability).

   [*The guidelines presented in the next section for the first task order provide insight into the types of deliverables that should be specified.*]

6. After the incremental deliverables produced under the first task order are evaluated, give careful consideration as to whether a second task order, similar in structure to the first one, should be negotiated and initiated before fully committing to the mining of legacy assets via follow-on task orders.

7. If the results are favorable, negotiate and fund additional task orders to have the contractor complete the mining of assets commensurate with the findings and recommendations reported in the first task order.

It should be noted that the government always has the right to terminate a contract either on the basis of *termination-for-convenience*[2] or *termination-for-default*,[3] but in this proposed

[2] *Termination-for-convenience* or "T for C" is invoked when the objective is determined to be unachievable due to problems that have arisen (and which the government has contributed to). In these cases, the government has to reimburse the contractor for the effort that has already been expended.

risk mitigation scenario the government doesn't need to terminate the contract should a problem arise. The government is only obligated to fund the minimum task order that the contract specifies. This minimum can be defined in terms of some minimum dollar amount, a number of discrete tasks and deliverables, or hours of effort that is commensurate with the program's risk mitigation approach. The objective in having the contractor perform the initial task-order is to allow the contractor to conclusively demonstrate their ability to perform critical tasks and mitigate high risk items without requiring the government to irrevocably commit, up-front, to one course of action.

Even if the results of the first task order indicated that it was not cost effective or practical to mine the assets, the contractor's services (and newly developed legacy system software expertise, which is a premium commodity) may be able to be put to good advantage. This can be accomplished by scoping the original ID/IQ task order contract to include the development of *new core assets* as an option that can be elected at the discretion of the government. Moreover, this option could be invoked even if the mining is partially, or wholly, successful, as there is certain to be a need for new asset development as the product line matures. As long as the contractor's performance is credible during the first task order, this arrangement would certainly be a "win/win" situation for both the government and the contractor.

### 4.4 Guidelines for Task Order Structuring

In addition to its use in structuring and monitoring a contract, concepts from OAR can help to guide the structuring of the pivotal tasks that are to be included in the first and second task orders. The specific guidelines for structuring these tasks[4] to facilitate decision making include the following:

1. Require the winning contractor to use the process[5] described in its proposal to explore and analyze a representative portion of the legacy system software and extract "as built" software design information corresponding to the three levels of design abstraction (i.e., code,[6] function, and architecture) outlined in OAR.

   [*The DoD contracting organization will identify the representative portion of the legacy system software to be mined by the contractor at the time of contract award.*]

---

[3] *Termination-for-default* or "T for D" can be invoked if it has been determined that the contractor can't meet the requirements. The amount of money the contractor is entitled to may be negotiated at the discretion of the contracting officer but it will not exceed the cost of the effort that has already been expended.

[4] An appropriate adaptation of the task requirements that are described is to be included in the SOW for the first task order.

[5] By "process" we mean the specific and enactable process for exploring, analyzing, and adapting legacy software that the offeror will be required to describe in its proposal for mining core assets.

[6] The degree to which the code level applies is questionable due to the fact that the emphasis is on large-grain software reuse. However, there may be special conditions that warrant changes at the code level that should not necessarily be ruled out.

2. Require the contractor to identify candidate assets for mining and to describe the functionality they would provide, the nature of the existing legacy interfaces, and any discovered architectural mismatches with the designated product line architecture.

3. Require the contractor to identify the specific types of transformations that would be required at the code structure (if applicable), function level, and architectural level to adapt the candidate assets so that they could be used as product line core assets.

4. Require the contractor to identify the overriding technical issues at each level of design abstraction, their potential impact, and the major tradeoffs that are involved in mining and adapting the legacy assets.

5. Require the contractor to identify the costs, risks, and schedule implications of performing these transformations to rehabilitate the legacy assets into product line core assets.

   [*An important aspect of these estimates is that they will be derived in a manner that is consistent with the quality principles [Deming 86] advocated by W. Edwards Deming by virtue of the fact they answer his famous quip, "By what process?"*]

6. Require the contractor to compare the mining/adaptation effort with a new development effort[7] and make recommendations, on an asset by asset basis, as to which are the most prudent in terms of risk, cost schedule, and quality attributes.

   [*The DoD contracting organization will identify any parameters corresponding to quality attributes (e.g., performance, security, and interoperability) at time of contract award.*]

7. Consider requiring the contractor to implement (i.e., mine and adapt) a small number of candidate assets at the function and architectural levels (commensurate with the transformations identified in the earlier steps) to convincingly demonstrate large grain reuse through wrapping or architectural adaptation.

   [*This option could be accomplished by separately negotiating it as the second task order under the ID/IQ contract.*]

These requirements would be provided in the form of guidance in the Proposal Preparation Instructions that are part of the RFP to ensure each offeror's proposal considers these aspects in describing their particular process. Each offeror's proposed mining and adaptation process would be evaluated as part of source selection. In addition, as part of the source selection process, offerors may be required to demonstrate any tool sets (commercial off the shelf [COTS] or proprietary) that their proposed processes are dependent on and/or provide examples of process artifacts described in their proposals.

[7] The contractor should describe to what extent, if any, the new development effort leverages the existing code structure level design information.

# 5 Summary and Conclusions

In this note we discussed the role of mining and adapting legacy system assets for use in product lines. We then outlined how insights from the OAR model can be applied to this problem. Building on these insights we developed a set of guidelines for using OAR concepts in mining legacy assets for a product line within a DoD acquisition setting. The guidelines are suitable for use by any DoD organization contemplating having a contractor mine their legacy systems to extract and adapt elements of the software for use as product line core assets.[8]

[8] The Product Line Systems Program would be interested in collaborating with a DoD activity in adapting these guidelines to their specific needs and assisting them in the preparation of an appropriate RFP.

## References


**Bass 99** Bass, L. & Kazman, R. *Architecture-Based Development* (CMU/SEI-99-TR-007, ADA366100). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. Available WWW: <URL: http://www.sei.cmu.edu/publications/documents/99.reports/99tr007/99tr007abstract.html>.

**Bergey 99a** Bergey, J.; Smith, D.; Weiderman, N.; & Woods, S.N. *Options Analysis for Reengineering (OAR): Issues and Conceptual Approach* (CMU/SEI-99-TN-014). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. Available WWW: <URL: http://www.sei.cmu.edu/publications/documents/99.reports/99tn014/99tn014abstract.html>.

**Bergey 99b** Bergey, J.; Fisher, M.; & Jones, L. *The DoD Acquisition Environment and Software Product Lines* (CMU/SEI-99-TN-004). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. Available WWW: <URL: http://www.sei.cmu.edu/publications/documents/99.reports/99tn004/99tn004abstract.html>.

**Carriere 99** Carriere, S.J.; Woods, S.G.; & Kazman, R. "Software Architecture Transformation," 13–23. *Proceedings of the Sixth Working Conference on Reverse Engineering* (WCRE '99), Atlanta, GA, October 6–8, 1999. Los Alamitos, CA: IEEE Computer Society, 1999.

**Clements 99** Clements, P. & Northrop, L.M. *A Framework for Product Line Practice* [online]. Software Engineering Institute, Carnegie Mellon University, 2000. <URL: http://www.sei.cmu.edu/activities/plp/framework.html.>

**Deming 86** Deming, W. Edwards. *Out of Crisis.* Cambridge, MA: Massachusetts Institute of Technology, Center for Advanced Engineering Study, 1986

**DoD 98** United States Department of Defense. *Defense Federal Acquisition Regulations Supplement,* 1998 Edition [online]. Available WWW: <URL: http://farsite.hill.af.mil/vfdfara.htm>. August 17, 1998.

**Kazman 97** Kazman, R. & Carriere, S.J. *Playing Detective: Reconstructing Software Architecture From Available Evidence*. (CMU/SEI-97-TR-010, ADA330928), October 1997, Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997. Available WWW: <URL: http://www.sei.cmu.edu/publications/documents/97.reports/97tr010/97tr010abstract.html>.

# Feedback and Contact

## SEI Technical Notes on Reengineering for Product Lines

Comments or suggestions about this document or the series of technical notes on *software reengineering for product lines* are welcome. We want this series to be responsive to the needs of DoD and government personnel involved in aspects of reengineering and acquiring software product lines. To that end, comments concerning this technical note, inclusion of other topics, or any other issues or concerns will be of great value in continuing this series. Comments or suggestions should be sent to

Linda Northrop, Director
Product Line Systems Program
lmn@sei.cmu.edu
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



| 1. AGENCY USE ONLY (LEAVE BLANK) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | March 2000 | Final |

**4. TITLE AND SUBTITLE**

Guidelines for Using OAR Concepts in a DoD Product Line Acquisition Context

**5. FUNDING NUMBERS**

**6. AUTHOR(S)**

John Bergey, Dennis Smith

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

**8. PERFORMING ORGANIZATION REPORT NUMBER**

CMU/SEI-2000-TN-004

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

HQ ESC/DIB
5 Eglin Street
Hanscom AFB, MA 01731-2116

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12.A DISTRIBUTION/AVAILABILITY STATEMENT**

Unclassified/Unlimited, DTIC, NTIS

**12.B DISTRIBUTION CODE**

**13. ABSTRACT** (MAXIMUM 200 WORDS)

Many DoD organizations are considering product line initiatives as a means of overcoming the issues of quality, cost and schedule inherent in a “one-at-a-time” system development or acquisition paradigm. Because a product line approach revolves around the creation of a comprehensive set of core assets, mining and adapting (i.e., reengineering) existing legacy system assets can offer significant leverage. By mining and adapting these assets, an organization can exploit the proven capabilities of existing systems, reduce the funding required, and develop/acquire new systems of higher quality within a shorter time frame.

This technical note focuses on providing guidance for DoD organizations for mining legacy systems to obtain core assets that will fit into a previously defined software architecture for a product line. We explain how a insights from a conceptual model, Options Analysis for Reengineering (OAR), can be used in an acquisition context to provide the government with an approach for obtaining greater insight and understanding into a contractor’s proposed technical reengineering approach.

**14. SUBJECT TERMS**

acquisition, Department of Defense, policies, product lines, software acquisition, regulations, mining legacy assets, reengineering

**15. NUMBER OF PAGES**

16 pp.

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |