Carnegie Mellon
Software Engineering Institute

Application of Options Analysis for Reengineering^{SM} (OAR^{SM}) in a Lead System Integrator (LSI) Environment

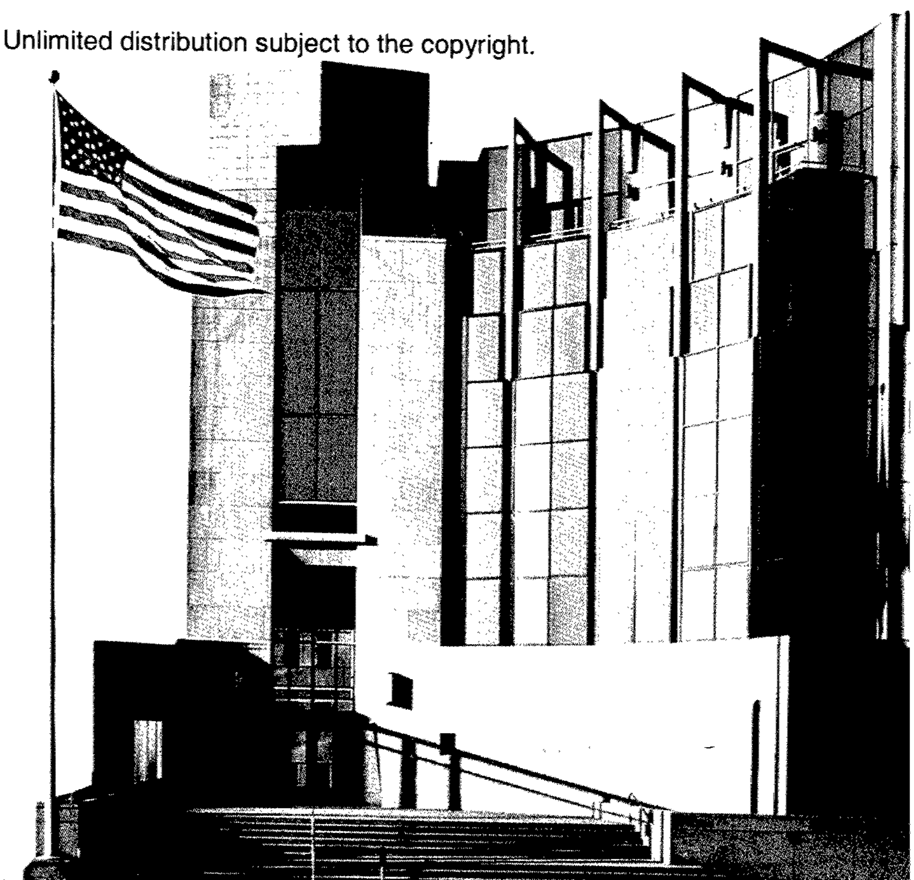
John Bergey
Liam O'Brien
Dennis Smith

March 2003

Product Line Practice Initiative

DISTRIBUTION STATEMENT A
Approved for Public Release
Distribution Unlimited

Technical Note
CMU/SEI-2003-TN-009



Application of Options Analysis for Reengineering^SM (OAR^SM) in a Lead System Integrator (LSI) Environment

John Bergey
Liam O'Brien
Dennis Smith

March 2003

Product Line Practice Initiative

Technical Note
CMU/SEI-2003-TN-009

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).

Contents

List of Figures

List of Tables

Acknowledgements

Options Analysis for Reengineering℠ (OAR℠) was developed as part of the program plan of the Product Line Practice Initiative under the leadership of Linda Northrop.

The work has received additional support from the Acquisition Support Program (ASP) as an acquisition pilot, so OAR could be tailored for use in a specific acquisition context. This support has enabled the baseline version of OAR to be adapted and piloted for a lead system integrator context. A pilot is currently being conducted on several projects with the Department of Army to understand the effectiveness of the revised OAR and to enable the wider use of the method to obtain more credible estimates of reuse.

The authors thank Reed Little and Grace Lewis for insightful comments on an earlier draft; these comments have been incorporated in this final version.

Abstract

Because most organizations have a substantial legacy base of existing software assets, few development efforts start from scratch. However, there has not been a systematic way to identify components for reuse or to understand the types of changes that would be required for insertion into a software product line architecture or a new software architecture.

Options Analysis for Reengineering^{SM} (OAR^{SM}) is an approach for making decisions on mining software assets. Mining involves rehabilitating parts of an old system for use in a new system. OAR identifies potential reusable components and analyzes the changes that would be needed to rehabilitate them for reuse within a software product line or new software architecture. OAR also provides an analysis of mining options, as well as the cost, effort, level of difficulty, and risks associated with each option. Recently, OAR has been applied to help a lead system integrator (LSI) make effective decisions on reuse. An LSI is the agent for an organization that is responsible for acquiring a large software-intensive system or system of systems. This note describes the use of OAR to guide decision making on mining assets within an LSI context, referred to as LSI OAR.

1 Introduction

Options Analysis for Reengineering[SM] (OAR[SM]) is an approach for making decisions on mining software assets [Bergey 01, Smith 02]. Mining involves rehabilitating parts of an old software system for use in a product line or new system. OAR identifies potential reusable components and analyzes the changes that would be needed to rehabilitate them for reuse within the target product line or software architecture. In addition, OAR provides an analysis of mining options, as well as the cost, effort, level of difficulty, and risks associated with each option.

OAR was initially developed and piloted for organizations that own both the legacy assets and the target system. Recently, it has been applied to help a lead system integrator (LSI) make effective decisions on reuse. An LSI is an agent with the authority to acquire and integrate assets from a variety of potential system suppliers on behalf of an organization that is acquiring a complex software-intensive system. The LSI has the authority to contract with and manage other suppliers on behalf of the acquirer.

A primary task of the LSI is to determine early in the integration cycle whether required software assets can be mined from existing assets, can be purchased as commercial off-the-shelf (COTS) components, or need to be developed from scratch. This note provides an overview of the use of OAR to guide decision making on mining assets within an LSI context, referred to as LSI OAR. In an actual engagement using LSI OAR, the team uses a set of data templates and execution templates to guide each of the activities and tasks.

Section 2 describes the background for OAR. Section 3 outlines the baseline application of OAR. Section 4 describes LSI OAR, its contrasts with the baseline OAR, and its use. Section 5 provides a summary and conclusions.

2 Background

Since most organizations have a substantial legacy base of existing software assets, few development efforts start from scratch, whether they are for a software product line or a new software-intensive system. However, until recently, there has not been a systematic way to identify components that are suitable for reuse, or to understand the types of changes that would be required for insertion into a product line or new software architecture [Müller 00]. In most cases, the only options available have been to undergo the costly and high-risk process of reengineering an entire system, to use an ad hoc reuse approach, or to simply build the new system from scratch.

Several researchers have outlined methods of rehabilitating components [Sneed 98, DeLucia 97]. Work has also been performed to identify risks in reengineering projects [Sneed 99]. However, previous work has not provided guidance on how to decide which components are viable candidates for mining or how to determine the effort, cost, and risks of a mining effort. As a result, projects often defer mining decisions indefinitely or base them on nebulous data. Doing this results in increased risk, increased problems to address, as well as missed opportunities. The OAR method addresses this need by establishing a systematic approach to decision making for mining software assets.

The OAR method is based on the premise that the costs and potential benefits of software reuse can be determined only in the context in which the software assets will actually be reused. As a result, reuse decisions need to be specific to the stated mission, business drivers, the target software architecture, and the specific component needs of the target system. The method needs to be systematic and credible enough to support the crucial early decisions on reusing legacy assets that may determine the success of the new system or product line.

In the baseline application of OAR, the assets to be reused are owned and managed by the same organization that is developing the new system or product line. In addition, there is some convergence of interest between the legacy group and the group responsible for developing the new system or product line.

In an LSI environment, the legacy group is part of an external supplier organization that is offering components to the LSI for integration into the target system. The supplier organization's goal is to maximize the size of its contract, while the LSI organization's goal is to understand the best fit of the offerings of several suppliers to the target system.

The baseline application of OAR has been modified to enable the LSI to make objective decisions between the offerings of different potential suppliers.

3 Baseline Version of OAR

The baseline version of OAR consists of five major activities, each with a set of tasks and subtasks that enable it to meet its goals. All tasks have execution templates to guide the process, and many have data templates that provide a starting point for creating customized data to support the information needs of the particular task. Each activity has a set of entry and exit criteria. The OAR analysis team (consisting of OAR experts, legacy-systems experts, and target-system experts) works within the client organization in a collaborative effort to perform the analysis.

The five major activities are described below:

1. *Establish the Mining Context (EMC)* establishes an understanding of the organization's product line or new single-system needs, legacy base, and expectations for mining legacy components. During this activity, a baseline of the goals is developed, along with the expectations for the mining project and the component needs that mining is to address. In addition, the programmatic and technical drivers for making decisions are determined, and a set of potential candidate components for mining is selected.

2. *Inventory Components (IC)* identifies the legacy-system components that can potentially be mined for use as product line or new single-system components. In this activity, the characteristics of the components' needs and screening criteria are identified. These screening criteria are used as a basis for evaluating legacy components, and those components that do not meet the criteria are screened out. This activity results in an inventory of candidate legacy components that fulfill components' needs.

3. *Analyze Candidate Components (ACC)* analyzes the candidate set of legacy components to evaluate their potential use as product line or new single-system components. During this activity, additional screening on the candidate component is performed, and the types of changes that are required to mine each candidate component are identified.

4. *Plan Mining Options (PMO)* develops alternative options for mining, based on schedule, cost, effort, risk, and resource considerations. During this activity, a final screening of candidate components is performed, and the impacts of different aggregations of components are analyzed.

5. *Select a Mining Option (SMO)* selects the mining option or combination of options that can best satisfy the organization's goals by balancing programmatic and technical considerations. Each mining option is evaluated, and the optimal option or combination of options is selected. A summary report and justification for the selected option are prepared.

Figure 1 illustrates the five OAR activities for the baseline version of OAR.

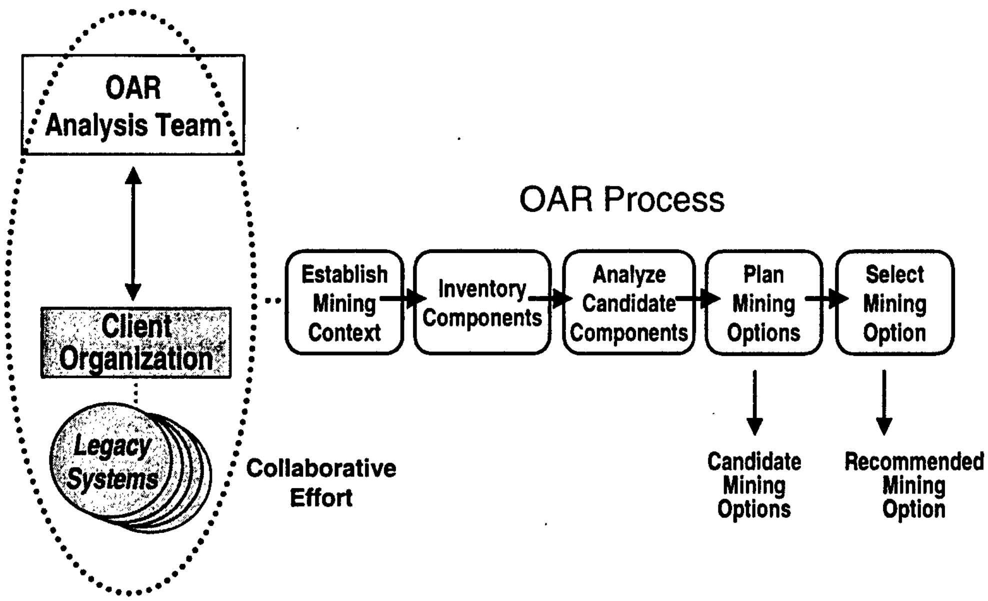


Figure 1: Baseline Application of OAR

4 LSI OAR

An LSI is the agent for an acquirer. For example, a government organization may be acquiring a complex software system or system of systems that includes a number of COTS components and reusable components from other systems. The primary engineering task is to develop an architecture for the entire system and then to assemble and integrate the components. Since the LSI relies extensively on assets that were initially developed by the individual system suppliers, it is crucial to accurately estimate the reuse potential of the existing assets. Without a disciplined approach such as OAR, which looks at the specific types of changes that must be made to an asset when it is placed in a new context, estimates from suppliers are essentially guesses that have unacceptable risk.

Two factors need to be considered when tailoring OAR for an LSI situation:

1. With the baseline application of OAR, all the work is performed within the same organization. In the case of an LSI OAR, there needs to be a division of labor between the supplier and the LSI, with the roles of each clearly defined.
2. Since the LSI and suppliers have different motivations, it is important to enable the LSI to objectively analyze the credibility of the data and recommendations provided by the suppliers.

Figure 2 illustrates how the division of labor is accomplished. The LSI is responsible for the first activity, EMC, which identifies the target architecture, the component needs, and the screening criteria. The supplier is responsible for the detailed technical analysis activities, including IC, ACC, PMO, as well as a new activity, Recommend Mining Option (RMO). The specific tasks of the analysis steps are modified to account for the fact that the supplier is making a recommendation to the LSI and that the supplier does not have the authority to make a decision. Guided by a new activity—Evaluate Mining Options, (EMO)—the LSI makes decisions on the recommendations provided by the suppliers.

The analysis provided by the suppliers may also affect the LSI from a broader perspective. An understanding of the specific options that are available may influence decisions on the target architecture. For example, one of the early pilots of LSI OAR provided substantially different conclusions on the potential reuse of assets depending on different assumptions about the architecture. The LSI used this data to refine its initial architecture to take advantage of the availability of the legacy assets.

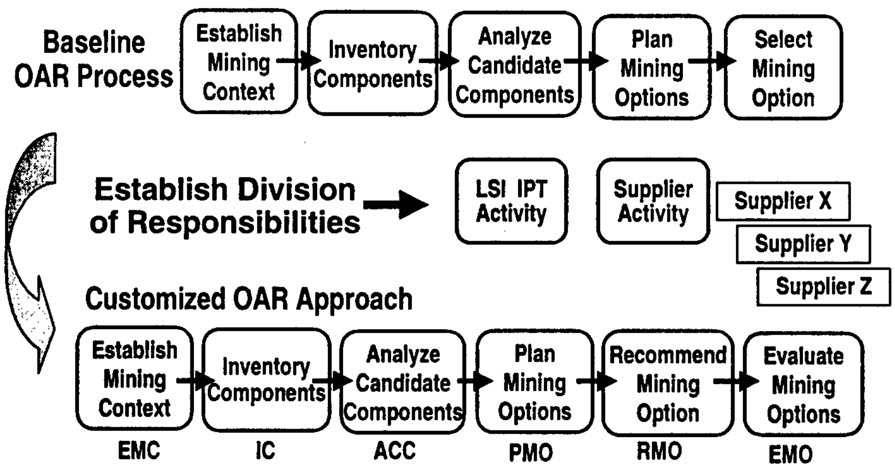


Figure 2: Overview of LSI OAR

4.1 Phases of LSI OAR

The LSI OAR process is implemented in two phases: an Initial Mining Analysis (Phase 1) and an optional In-Depth Mining Analysis (Phase 2). The first phase provides rough ("ballpark") estimates in a relatively short period of time. In some cases, the Phase 1 estimates may be sufficient for making a decision.

However, if the variability of the estimates is high, the risks of making the required changes are high, or the LSI wishes to make a further down selection among the suppliers, a Phase 2 analysis may be used to provide more detailed data and a higher level of confidence for decision making. During a Phase 2 analysis, the suppliers perform more detailed analyses before the LSI makes final decisions on specific components. If changes to the target architecture are made as a result of the Phase 1 analysis based on the availability of assets, these changes are incorporated as the baseline architecture for Phase 2.

The two-phase LSI OAR process is shown in Figure 3. Figure 3 distinguishes between those activities that are the responsibility of the LSI and those that are the responsibility of the supplier.

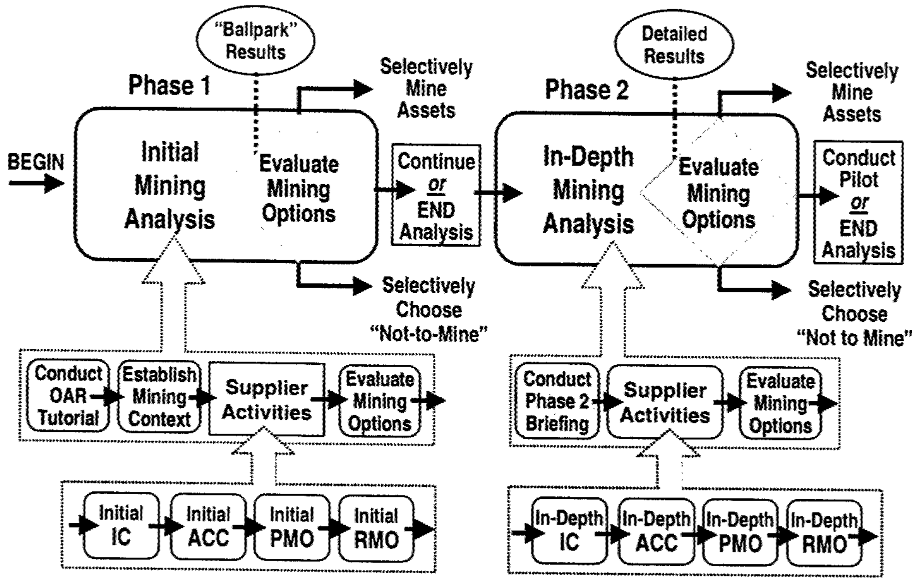


Figure 3: Details of LSI OAR by Phase

LSI OAR is currently being piloted for Phase 1. As a result, this note focuses on Phase 1 activities. Section 4.2 describes Phase 1 and provides some examples of how it can be applied. Section 4.3 provides an overview of Phase 2. Phase 2 activities will be described in greater detail in a future technical note after they are piloted.

4.2 Overview of Initial Mining Analysis (Phase 1)

The process for the Initial Mining Analysis is shown on the left side of Figure 3.

The LSI defines the overall mining need and provides a tutorial on the OAR process through the first two tasks: Conduct OAR Tutorial (COT) and Establish Mining Context (EMC). The participating suppliers next perform the core mining and analysis activities in which they Inventory Components (IC), Analyze Candidate Components (ACC), Plan Mining Options (PMO), and Recommend a Mining Option (RMO). The LSI then makes its decisions for the initial phase through Evaluate Mining Options (EMO).

The following subsections highlight some of the major tasks and provide examples of how they are used.

4.2.1 LSI Initial Tasks

To establish a common ground, the OAR tutorial provides an overview for all potential suppliers. Supplier participants include technical management and experts on the relevant legacy systems.

The LSI is responsible for the EMC activity. The LSI

- defines the goals and objectives for the effort, along with the expectations and drivers of the target-system stakeholders
- conducts a walkthrough of the new system to describe the technical requirements that set the context for the mining effort (This walkthrough includes a review of the software architecture, its components, and the available system documentation.)
- reviews the software component needs for the new system in detail with the suppliers to ensure a proper understanding of the technical requirements for the mining effort
- reviews the screening criteria and common component characteristics that will drive the mining effort
- identifies the output artifacts to be produced by each supplier upon completion of its mining analysis and the output artifacts the LSI will produce to document its evaluation of each supplier's recommended mining option(s)

4.2.1.1 Example of Goals and Objectives

The goals and objectives for the specific mining effort are defined by the LSI during EMC. Examples include the following:

- Develop the system in the most cost-effective manner possible consistent with a product line approach.
- Minimize the time needed to deploy the software system by using mined components to the maximum degree practical even if they need to be replaced in the long term.
- Deliver the software to system test within a one-year period.
- Do not mine legacy components if the cost exceeds 60% of new development cost.
- Eliminate reliance on software components written in archaic languages and obsolete software support systems.
- Provide a Web-based user interface.

4.2.1.2 Example of Component Needs

The component needs that are defined by the LSI represent an important starting point for the analyses that the suppliers perform. Software component needs may include

- required functionality
- required user interface features and constraints

- component-level quality attribute needs, such as security, reliability, and memory usage throughput
- technology features and constraints, such as application program interfaces (APIs)
- software engineering environment tools

4.2.1.3 Example of Screening Criteria

Each analysis will identify a different set of screening criteria for evaluating the candidate components. These characteristics are identified by the LSI during EMC.

Once the high-level component needs are identified, a set of required component screening characteristics are defined for the mining effort. Each component is analyzed based on two types of characteristics: common component characteristics and component-specific characteristics.

Common component-screening characteristics are those that are analyzed for each relevant component. These may include

- common interface constraints
- technical features and constraints, such as programming language, complexity, cohesion, and coupling
- technology features and constraints, such as specific required standards

In addition, there may be component-specific characteristics where different types of components may have different types of needs and constraints. These may include specific functionality needs, specific interface constraints, and specific component-level quality attribute needs, such as security.

4.2.2 Supplier Activities

Once the LSI establishes the context, the suppliers perform the following core mining activities:

- Inventory Components to develop an initial set of components that could potentially meet the component needs and screening criteria identified by the LSI.
- Analyze Candidate Components to examine the candidate components and determine the types of changes that would be required to rehabilitate the legacy assets for use in the target architecture, as well as the estimated cost and risk of making the changes.
- Plan Mining Options to aggregate candidate components into different options that may satisfy the needs of the LSI. The options are then analyzed in terms of cost, effort, difficulty, and risk.

- Recommend a Mining Option to summarize the supplier's recommended best option to the LSI.

4.2.2.1 Example of Component Table

As outlined in Section 4.2.1, the LSI defines the relevant component needs and screening criteria during EMC. The supplier then analyzes a set of potential components based on these criteria.

During the technical analysis, each supplier produces a component table to summarize the relevant data that are captured for each component need. The table is filled out cumulatively throughout the analysis activities.

The component table will differ for each analysis because the screening criteria are different. Tables 1 and 2 show examples of parts of a component table for a sample analysis. This analysis is examining potential components from a satellite tracking system.

Table 1: Example of Component Table: Common Component Characteristics

		Required Common Component Characteristics			
New System Component Need	**Legacy-System Software Components**	**Programming Language**	**Source Size Lines of Code**	**Number of Modules**	**Structured Code**
Simulation Gateways	Sim1 Gateway Sim2 Gateway	C C	2,257 4,877	11 57	Yes
Message Gateways	C-Source	C	4,186	45	Yes
Truth Model	Objects Event	Fortran Fortran	5,202 4,412	65 39	Yes Yes
Event-Tracking Subsystem and Message Generation	Event Track	C	5,417	89	Yes

In Table 1, the first column shows that four component needs for the target system have been identified: simulation gateways, message gateways, a truth model, and an event-tracking subsystem. The second column lists the components that are relevant for each need. The other columns identify some of the required, common, component-screening characteristics that have been identified by the LSI, including programming language, source lines of code, number of modules, and whether the code is structured.

Table 2 illustrates the rehabilitation characteristics of a sample component table, including the level of change required, software support required, level of difficulty, and mining effort in months.

Table 2: Example of Component Table: Rehabilitation Characteristics

		Rehabilitation Characteristics			
New System Component Need	**Legacy-System Software Components**	**Level of Changes Required**	**Support Software Required**	**Level of Difficulty**	**Mining Effort (MM)**
Simulation Gateways	Sim1 Gateway Sim2 Gateway	None	S&D[a] S&D	1 1	0.1 0.1
Message Gateways	C-Source	None	S&D	1	0.1
Truth Model	Objects Event	Minor: 10% Minor: 10%	S&D S&D	2 2	2.2 1.3
Event-Tracking Subsystem and Message Generation	Event Track	Major: 50%	S&D	4	8

[a] S&D: Script and data files

An abstract version of a component table is shown in Figure 4. This figure shows a more complete list of component characteristics and rehabilitation characteristics, as well as new development estimates and estimates for mining versus new development. An actual component table for any specific analysis will, of course, vary depending on the relevant characteristics.

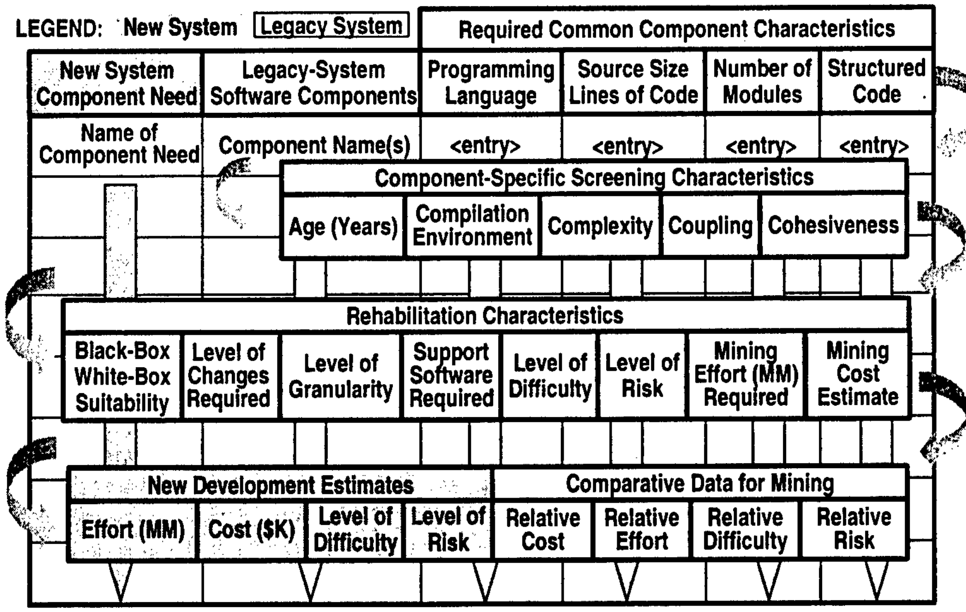


Figure 4: Abstract Example of a Component Table (Customized for Each Effort)

Figure 4 also shows the costs for performing the rehabilitation, based on the supplier's historical data or the experience of the software maintainers and other subject matter experts. These estimates are then compared to the cost of building the components from scratch, which is also based on the supplier's historical data.

4.2.3 LSI Decision

After each supplier presents the results of its mining analysis and its recommended options, the LSI evaluates all the options that have been presented by the different suppliers. During the EMO activity, the LSI analyzes the credibility of the suppliers' recommendations, based on adherence to the process, technical credibility, cost effectiveness of the proposal, and relationship of the supplier's proposal to those of other suppliers. At this time, the data from the suppliers are also provided to the LSI software architecture team to determine if changes to the target software architecture are warranted based on the availability of specific components.

4.2.4 Phase 1 Exit Criteria

Phase 1 provides a disciplined process to guide the suppliers in conducting a "ballpark" analysis of the reuse potential of their legacy assets. These analyses enable the LSI to make informed decisions and to choose between the different recommendations made by the suppliers.

The exit criteria that are required for the completion of Phase 1 are listed below.

1. Each supplier
   - produces a *Summary Analysis Report* describing its analysis results and recommended mining option(s)
   - prepares and delivers a *Summary Out-Briefing Presentation* to the LSI summarizing its analysis results and recommended mining option(s) (The presentation should include the "ballpark" cost, schedule, and risk estimates for rehabilitating the legacy software assets for each mining option being recommended.)
2. The LSI
   - evaluates each supplier's findings and recommended mining option(s)
   - produces an overall *Summary Evaluation Report* for LSI management describing the recommended selections for mining and rehabilitating legacy software assets based on all the suppliers' findings and recommendations
   - prepares and delivers an *Individual Out-Briefing Presentation* to each supplier summarizing the evaluation results and the proposed follow-on involvement of the supplier in mining and rehabilitating legacy software assets

4.2.5 Potential Outcomes of Phase 1

The three potential outcomes of Phase 1 are provided below:

1. It is definitely not beneficial to mine certain components. As a result, work should be planned to develop these software components from scratch or acquire them from other sources.
2. It is definitely beneficial to mine certain components, and the in-depth Phase 2 analysis can be skipped for these components.
3. Certain components have the potential for mining, but more information is needed before a final decision can be made. An in-depth analysis is required to make a proper determination about the benefits and desirability of mining these components. This in-depth analysis may include an analysis of the target software architecture to determine if changes are warranted based on the availability of a specific set of components. In this case, a Phase 2 analysis will be required.

Based on the results of Phase 1, the more detailed analysis of Phase 2 may be required.

4.3 Phase 2 Overview

As previously mentioned, Phase 2 has not yet been piloted. However, a brief overview is presented below.

Phase 2 has almost the same set of activities as Phase 1. However, the emphasis and specific tasks will differ. The Phase 2 process is illustrated on the right side of Figure 3. The differences between Phase 1 and Phase 2 are highlighted in the following subsections.

4.3.1 Phase 2 Briefing

The Phase 2 briefing provides a common ground for all the suppliers. It covers the following:

- selective disclosure of the Phase 1 decisions (The initial analysis results of each supplier should be kept confidential.)
- an overview of the In-Depth Mining Analysis core activities (IC, ACC, PMO, RMO, EMO)
- any changes to the mining context, software architecture, software component needs, or other aspects that could affect the Phase 2 effort
- any issues or concerns about the process or technical and programmatic aspects that govern the mining effort

4.3.2 Supplier Activities

After the briefing, the suppliers perform the detailed analysis of the relevant components, focusing on a detailed analysis rather than a "ballpark" estimate. For example, in Phase 1 there may have been a conclusion that a new interface would be required for a component and that a specific piece of functionality would need to be added. In Phase 2, the analysis specifies the details of the changes; determines any required constraints; determines any dependencies on middleware, scripts, or data files; and determines at a lower level the specific changes that must be made. This lower level of analysis will lead to a more precise estimate of the required effort. The cost estimates can be put into a formal costing model such as the Constructive Cost Model (COCOMO) II [Boehm 00]. They can also include projections for life-cycle costs.

4.3.3 Evaluate Mining Options

The detailed results are provided to the LSI. The LSI compares the supplier analyses according to previously selected weighting criteria, and makes decisions on each supplier's detailed proposal and on the impact of the cumulative set of suppliers' proposals for populating the target system.

5 Summary and Conclusion

The LSI OAR method provides a common process to ensure consistency of results. The process focuses on what activities need to be performed. Suppliers have the freedom to decide how best to perform the prescribed activities and tasks, and how to collect the needed data.

Work is allocated between the LSI and the supplier based on a natural division of responsibilities (i.e., who is in the best position to perform an activity). As a result, fewer LSI resources are required to complete the mining analyses, which results in reduced cost. The approach obtains early "ballpark" results to accelerate the decision-making process. It also provides for an in-depth analysis to enable the analysis results to be refined when needed.

An adaptation of the LSI OAR method has been piloted within a large-scale government program that relies heavily on reuse. The pilot has enabled more realistic estimates by suppliers and will be expanded for wide-scale application within that program.

References

[Bergey 01] Bergey, John; O'Brien, Liam; & Smith, Dennis. *Options Analysis for Reengineering (OAR): A Method for Mining Legacy Assets* (CMU/SEI-2001-TN-013, ADA 395201). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001. <http://www.sei.cmu.edu/publications/documents/01.reports/01tn013.html>.

[Boehm 00] Boehm, B.; Horowitz, E.; Madachy, R.; Reifer, D.; Clark, B.; Steece, B.; Brown, A. W.; Chulani, S.; & Abts, C. *Software Cost Estimation With COCOMO II*. Upper Saddle River, NJ: Prentice Hall, 2000.

[DeLucia 97] De Lucia, A.; Di Lucca, G A.; Fasolino, A. R.; Guerra, P.; & Petruzzelli, S. "Migrating Legacy Systems Towards Object-Oriented Platforms," 122-129. *Proceedings of the International Conference on Software Maintenance*. Bari, Italy, Oct. 1-3, 1997. Los Alamitos, CA: IEEE Computer Society, 1997.

[Müller 00] Müller, H.; Jahnke, J.; Smith, D.; Storey, M. A.; Tilley, S.; & Wong, K. "Reverse Engineering: A Roadmap," 47-60. *Proceedings of the International Conference of Software Engineering: The Future of Software Engineering*. Limerick, Ireland, June 4-11, 2000. New York, NY: ACM, 2000.

[Smith 02] Smith, Dennis; O'Brien, Liam; & Bergey, John. "Using the Options Analysis for Reengineering (OAR) Method for Mining Components for a Product Line," 316-327. *Proceedings of the Software Product Lines: Second International Conference (SPLC2)*. San Diego, CA, August 19-22, 2002. Berlin, Germany: Springer-Verlag, 2002.

[Sneed 98] Sneed, H. & Majnar, R. "A Case Study in Software Wrapping," 86-93. *Proceedings of the International Conference on Software Maintenance*. Bethesda, MD, Nov. 16-18, 1998. Los Alamitos, CA: IEEE Computer Society, 1998.

[Sneed 99] Sneed, H. "Risks Involved in Reengineering Projects," 204-211. *Proceedings of the Sixth Working Conference on Reverse Engineering*. Atlanta, GA, Oct. 6-8, 1999. Los Alamitos, CA: IEEE Computer Society, 1999.

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	March 2003	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Application of Options Analysis for Reengineering^{SM} (OAR^{SM}) in a Lead System Integrator (LSI) Environment	F19628-00-C-0003

6. AUTHOR(S)

John Bergey, Liam O'Brien, Dennis Smith

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2003-TN-009

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116	

11. SUPPLEMENTARY NOTES

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

13. ABSTRACT (MAXIMUM 200 WORDS)

Since most organizations have a substantial legacy base of existing software assets, few development efforts start from scratch. However, there has not been a systematic way to identify components for reuse or to understand the types of changes that would be required for insertion into a product line architecture or a new software architecture.

Options Analysis for Reengineering^{SM} (OAR^{SM}) is an approach for making decisions on mining software assets. Mining involves rehabilitating parts of an old system for use in a new system. OAR identifies potential reusable components and analyzes the changes that would be needed to rehabilitate them for reuse within a software product line or new software architecture. OAR also provides an analysis of mining options, as well as the cost, effort, level of difficulty, and risks associated with each option. Recently, OAR has been applied to help a lead system integrator (LSI) make effective decisions on reuse. An LSI is the agent for an organization that is responsible for acquiring a large software-intensive system or system of systems. This note describes the use of OAR to guide decision making on mining assets within an LSI context, referred to as LSI OAR.

14. SUBJECT TERMS	15. NUMBER OF PAGES
lead system integrator, LSI, Options Analysis for Reengineering, OAR, software architecture, software product line architecture	28

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