Mission Success in Complex Environments (MSCE)

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213
Mission Success in Complex Environments (MSCE) Project

Part of the SEI Acquisition Support Program (ASP), the MSCE Project develops methods, tools, and techniques for

- Advancing the state-of-the-practice for risk management
- Assuring success in complex, uncertain environments

The project builds on more than 17 years of SEI research and development in risk management.

- Continuous Risk Management for software-development projects
- Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE®) for organizational security
Widespread Use of Risk Management

Most programs and organizations implement some type of risk management approach when developing and operating software-intensive systems.

- Risk management plan
- Processes
- Tools

However, preventable failures continue to occur.

- Uneven and inconsistent application of risk-management practice
- Significant gaps in risk-management practice
- Ineffective integration of risk-management practice
- Increasingly complex management environment
## Changing Risk Paradigm

<table>
<thead>
<tr>
<th>From Traditional Paradigm</th>
<th>To New Paradigm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tactical analysis that produces point mitigation solutions</td>
<td>Systemic analysis that produces strategic mitigation solutions</td>
</tr>
<tr>
<td>Failure-oriented (“playing not to lose”)</td>
<td>Success-oriented (“playing to win”)</td>
</tr>
<tr>
<td>Narrow tradeoff space based on type of risk (e.g., program, security)</td>
<td>Broad tradeoff space based on mission and objectives</td>
</tr>
<tr>
<td>Applicable to a specific life-cycle phase and a single group or team</td>
<td>Applicable across the life cycle and supply chain (multi-enterprise/system environments)</td>
</tr>
<tr>
<td>Stand-alone management practice</td>
<td>Integrated with program and organizational management practices</td>
</tr>
<tr>
<td>Bureaucratic and time-intensive</td>
<td>Practical, straightforward, and easy to apply</td>
</tr>
</tbody>
</table>
Tactical and Systemic Approaches

Systemic View of Risk

Tactical View of Risk

Potential Event → Consequence
Condition

Potential Event → Consequence
Condition

Consequence
Condition

Potential Event
Condition

Impact on Objectives
What
A suite of risk-based methods and guidance for managing systemic risk across the life cycle and supply chain

Benefits
Focused on achieving operational success
Enables continuous management of risk
Applicable across all life-cycle phases
Designed for multi-enterprise, multi-system environments
Provides a means of analyzing risk in relation to management models, frameworks, and standards
Mosaic: Focus on Assessment

Every organization has preferred management practices.

The foundation of the Mosaic approach is a suite of methods for assessing risk continuously.

Mosaic also provides guidance for leveraging existing management practices to develop, implement, and track risk mitigation plans.
Mosaic Assessments

Mosaic provides a suite of methods for assessing risk

Mosaic assessments are modular in design

Driver identification and analysis provide a common front end for multiple back-end analyses
Mosaic: *Driver-Based Assessment*

A driver is a factor that has a strong influence on the eventual outcome or result.
Driver Framework

Driver Categories

- Objectives
- Preparation
- Execution
- Environment
- Resilience
- Result

The driver framework is a common structure for classifying a set of drivers.
**Driver Attributes**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A concise label that describes the basic nature of the driver</td>
<td>Process</td>
</tr>
<tr>
<td>Success State</td>
<td>A driver exerts a positive influence on the outcome</td>
<td>The process being used to develop and deploy the system is sufficient.</td>
</tr>
<tr>
<td>Failure State</td>
<td>A driver exerts a negative influence on the outcome</td>
<td>The process being used to develop and deploy the system is insufficient.</td>
</tr>
<tr>
<td>Category</td>
<td>The category to which the driver belongs</td>
<td>Preparation</td>
</tr>
</tbody>
</table>
## Basic Set of Drivers for Software Development

1. Program Objectives  
2. Plan  
3. Process  
4. Task Execution  
5. Coordination  
6. External Interfaces  
7. Information Management  
8. Technology  
9. Facilities and Equipment  
10. Organizational Conditions  
11. Compliance  
12. Event Management  
13. Requirements  
14. Design and Architecture  
15. System Capability  
16. System Integration  
17. Operational Support  
18. Adoption Barriers  
19. Operational Preparedness  
20. Certification and Accreditation
Driver Analysis

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Is the process being used to develop and deploy the system sufficient?</td>
<td>![Likely no]</td>
</tr>
<tr>
<td>Consider: Process design; measurements and controls; process efficiency and effectiveness; acquisition and development life cycles; training</td>
<td></td>
</tr>
</tbody>
</table>

Driver questions are phrased from the success perspective. Probability is incorporated into the range of answers for each driver. The rationale for selecting an answer is recorded.
Integrating Tactical Data

A driver-based approach enables integration of tactical data.
A simple analysis provides insight into current conditions.
Primary Relationships among Driver Categories

Environment

Objectives

Resilience

Execution

Result

Preparation
Additional Analysis of Drivers

Drivers provide a foundation for program decision making.

A variety of back-end analyses can be used to analyze a set of driver values.

- Gap analysis
- Risk analysis
- Mission success analysis
- Mission assurance analysis
- Integrated risk and opportunity analysis
From Drivers to Risks

<table>
<thead>
<tr>
<th>Risk</th>
<th>Probability</th>
<th>Impact</th>
<th>Risk Exposure</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. The process being used to develop and deploy the system is insufficient.</td>
<td>High</td>
<td>Severe</td>
<td>High</td>
</tr>
</tbody>
</table>

Determined using results of driver analysis

Determined using standard risk analysis methods
Multi-Enterprise Environments: *Network of Objectives*
Multi-Enterprise Environments: Applying the Driver Framework

Assessing a distributed program requires examining

- Each individual group
- The end-to-end program
Mosaic Assessments: Application in Multiple Domains

Software acquisition and development programs
Process improvement
Mission assurance
Software assurance
Information technology management
Cyber security management
Critical infrastructure protection
Risk Management Framework -2

The Risk Management Framework is implementation independent.

- Defines risk management activities
- Does not specify how to perform those activities

The framework provides a

- Foundation for a comprehensive risk management methodology
- Basis for improving a risk management practice
Mosaic Portfolio - 1

Courses

- Risk Management Framework: Best Practices in Risk Management
- Introduction to Practical Risk Management
- Practical Risk Management: Framework and Methods

Workshops

- Risk Management Tailoring and Improvement Workshops

Course and Workshop Combinations
Evaluations

- Program Risk Evaluation
- Mission Success Evaluation
- Risk Management Framework Evaluation
- Custom Evaluation
Future Research

Metrics
Risk-based improvement
Modeling and simulation
For Additional Information

Christopher Alberts
Email: cja@sei.cmu.edu
Phone: 412-268-3045
Fax: 412-268-5758

Audrey Dorofee
Email: ajd@sei.cmu.edu
Phone: 412-268-6396
Fax: 412-268-5758

WWW http://www.sei.cmu.edu/msce/
U.S. mail Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890