Anteon™

ilTa
Learn Perform Mentor
Enterprise Architect
Consulting Solutions, Deployment and Training

Realizing CMMI® using Enterprise Architect and UML for Process Improvement

Jack Hunnicutt,
Anteon Corporation
www.anteon.com

Ramsay Millar,
integrate IT architects LLC
www.integrateITarchitects.com

CMMI®

ENTERPRISE ARCHITECT

Introduction

The growing number of organizations that have embraced software process improvement has moved the software engineering sector from the land of the unknown to a predictive and repeatable set of practices. This is possible in large part to following a proven model such as the Software Engineering Institute's (SEI) Capability Maturity Model® Integration (CMMI[SM]) and employing affordable software tools adaptable throughout the organization. While understanding that process improvement can be a very costly expense, process maturity can be realized for those small to mid-size organizations with fiscal constraints.

This paper addresses the approach and resulting benefits of employing UML to integrate work products with Sparx Systems' Enterprise Architect to achieve CMMI Maturity Level 2. These realized benefits include: centralized data management, automated level of effort estimation, documenting and tracking action items, resource management, requirements management/traceability, risks identification, and metrics collection.

The principles of CMMI assume an adequate and functioning infrastructure is in place although specifically not mandated by the model. Enterprise Architect provided a low-cost CASE tool environment, which integrates the activities of the various project teams such as design, technical development, functional analysis and software testing. Bringing the business process into a common work environment encourages interactions between stakeholders while standardizing processes across the project and organization.

Each of the seven CMMI Maturity Level 2 Process Areas are analyzed in this paper to identify the level at which the Specific Practices were achieved through Enterprise Architect. Based upon these proven capabilities, this paper documents the highlights of how an organization achieved a higher level of software development maturity based upon UML.

What does this mean to your organization? Organizations that take the approach of this paper can realize at a minimum; implementation of a low-cost data repository to include a requirements database, real-time resource monitoring with usable project estimates, and a common technical infrastructure enhancing cross-project communications. The end result is deployment of a standard set of processes with an emphasis on quality while reducing the cost of infrastructure.

Background

Historically, process improvement has been viewed as a hindrance to *real work*. Some have also referred to process improvement as a set of 'unnatural acts', which are performed merely for the sake of compliance. Management and the aptly named 'Process Champions' have struggled with implementing process change without impacting schedules and the normal day-to-day activities of software development. These misconceptions and obstacles are quickly fading thanks to tool-based deployment of CMMI.

Employing automated tools for process implementation is not a new concept but an evolving science of maximizing actual value realized through organizational adoption of workable solutions. Most organizations are successful to a degree before starting their process

improvement journey. The real challenge has been to surface the proven processes, develop processes for the weak areas, and institutionalize the process into the corporate culture while at the same time obtaining measurable results for process change. This is an expensive and almost overwhelming undertaking without a dynamic automated environment adaptable to each unique situation.

Although initially conceived as a language for software development; Unified Modeling Language (UML) is used to model a wide range of real world domains. UML is a standard that defines rules and notations for specifying business and software systems. UML is not only a tool for creating software systems, it is also a visual language for communicating, modeling, specifying and defining systems.

Tool Driven Management

The evolving software development industry has rapidly adopted tool solutions that allow large teams of IT staff to openly share artifacts using UML. Without knowing the full capability, we selected Enterprise Architect primarily as a modeling tool but quickly realized UML 2.0 could support other needs of our development team. Following close behind an intensive training class we began to view UML as a repository of data and work products that could be shared across the project or organization in a common environment. This really opened the door to experimenting with linking both internal and external files, eliminating redundancy and maximizing product reuse in addition to the ability to trace requirements from conception to production.

Wanting to capitalize on our newly found capabilities, the expectations we had of Enterprise Architect started to expand and eventually focused heavily on project management. Producing project estimates was one of the paramount reasons for our UML implementation, which was achieved successfully early on. But we were now asking, what about resources, status of resources and project status? This was the turning point of realizing that by employing tool driven management, process maturity could be achieved with limited infrastructure, across multiple domains without the need to increase the workforce or acquiring high-end development tools.

Strategy and Results

The decision to employ UML using Enterprise Architect as the project repository was based upon the need to implement new technology without substantial risks to schedule and cost, and to affect the change in a non-intrusive manner from the customer's perspective. The organizations business objectives drove the implementation strategy in regard to costs, training and schedule in addition to establishing an acceptable level of process enhancements (CMMI ML2). Based on these conditions, our successful deployment of UML and Enterprise Architect resulted in the developed of a process focused framework that achieved the following:

- Produce realistic and usable level of effort estimates. The tool has a very robust estimation feature within the Use Case Metrics. Based on Technical and Environmental Complexity Factors, estimates were easily modified and tuned. Historical data from prior

estimates were compared to actual milestones providing an opportunity to adjust the Use Case Metrics setting to produce increasingly more realistic estimates.

- Maintain visibility into project issues and concerns. Utilizing the incorporated Model Task and Model Issues, detailed tasks may be assigned to reflect responsibility and tracking. Additionally, project issues or concerns can be documented and tracked to closure. This eliminated the need for an external action item tracking system and greatly reduced the possibility of system issues loosing visibility.

- Document, categorize and track risks. The Project Management module provides the ability to identify risks at the Use Case level. Each identified risk includes a detailed description, category or type of risks and a weighted value.

- Improve development team communications and promote cross-team collaboration. The UML environment itself achieved this. All project personnel including functional, technical, testing and management accessed common data in real-time. With traditional methods one team would not have access to work products such as functional analysis until the analysis was completed. But now working in a common environment it was possible to obtain "early" reviews of work products promoting team collaboration and early defect identification.

- Capture and report project status for both real-time and historical purposes. Another strong feature of the Project Management module was Resource Allocation. Although short of an actual WBS, Enterprise Architect does track and report resource activities. This includes identification of the resource by name and role, start and end dates, percentage complete and input fields to record descriptions and historical data.

CMMI Practice Satisfaction

Satisfying the seven (7) Process Areas and associated Generic Practices of CMMI ML 2 is a significant challenge in any environment. We found that by utilizing the standard package of desktop applications (Word, Excel and email) and a configuration management system in concert with Enterprise Architect a vast majority of specific and generic practices could be achieved. Based on the Enterprise Architect features available during the winter of 2005, each Process Areas is analyzed in the tables below to show the level of fidelity achieved.

Fidelity Level:

High – Satisfied
Medium – Rely upon some external support
Low – Not supported within tool

Requirements Management (REQM)

Manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products. [i]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Manage Changes/Change History	High	Integrated Requirements Database
Bidirectional Traceability	High	Hierarchy Diagram
Traceability Matrix	Moderate	Hierarchy Diagram
Requirement Status/Impacts	High	Requirement Properties
Requirements Database	High	Supports 8 database types (i.e.DB2, MySQL, Oracle, SQL Server, Jet Databases)
Requirement Tracking	High	Status & Report Generation
Documented Commitments	Moderate	Recorded in narrative format

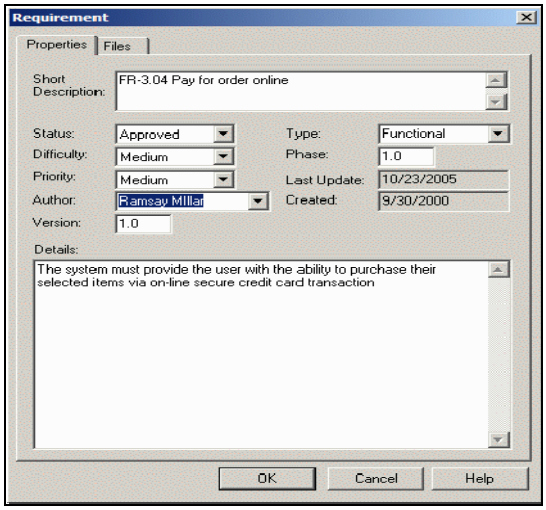

Figure 1 - Requirements Management. Requirements screen documents a requirement showing Status, Difficulty, Priority, Type, Phase, Author and supporting Details. Additionally, external files may be attached to provide supplemental requirement information.

Project Planning (PP)

Establish and maintain plans that define project activities. [ii]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Task/Work Package Descriptions	High	Resource Allocation
Work Breakdown Structure (WBS)	Low	API for third-party product
Determine Technical Approach	High	Use Case Properties
Size and Complexity	High	Use Case Metrics
Estimate Model/Attributes	High	Use Case Metrics
Life-Cycle Phasing	Moderate	Rely upon linked WBS
Estimate Effort & Cost	Moderate	Use Case Metrics
Project Schedule/Budget	Low	Rely on linked WBS
Risk Identification/Priorities	High	Project Management Module

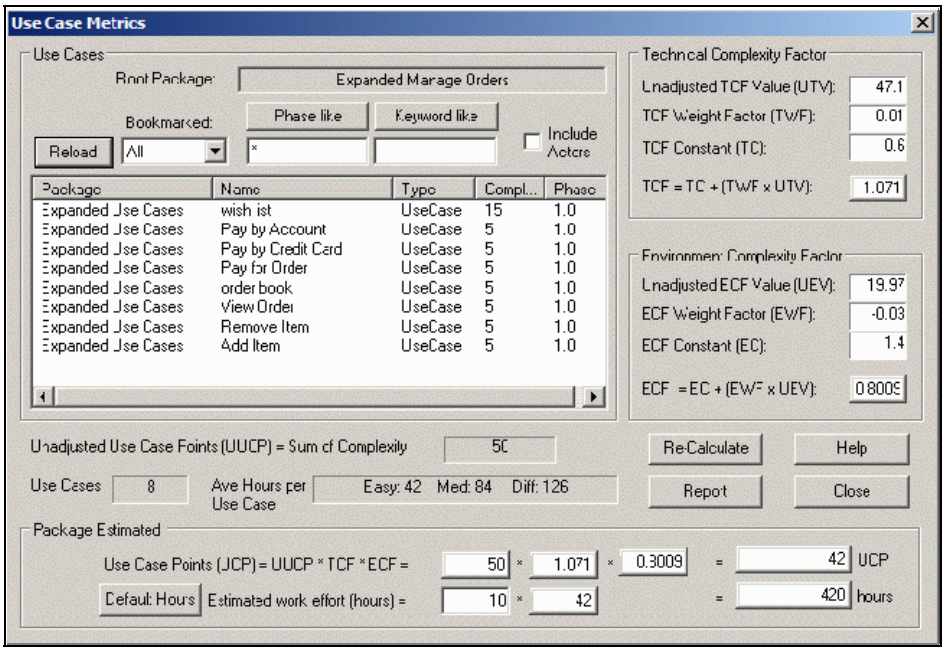

Figure 2 - Use Case Metrics – Estimate Effort based on hours derived from a Use Case sizing value and Technical and Environmental Complexity Factors.

Project Monitoring and Control (PMC)

Provide an understanding of the project's progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan. [iii]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Progress Monitoring/Reporting	High	Resource/Effort Reports
Actual vs. Estimates	Moderate	Rely upon linked or API to WBS
Monitor Attributes/Resources	High	Baseline Estimates
Risks Monitoring	High	Project Management Module
Monitor Stakeholder Involvement	Moderate	Resource Allocation
Milestone Reviews	Low	Rely upon linked or API to WBS
Progress Reviews	Moderate	Resource/Effort Reports
Issue Identification/Tracking	High	Project Management Risks

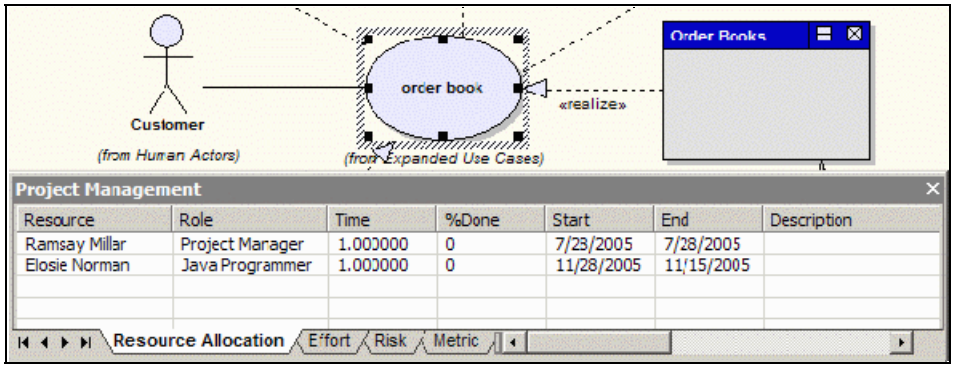

Figure 3 - Project Management Resource Allocation - Stakeholder responsibility is assigned and tracked in real-time.

Supplier Agreement Management (SAM)

Manage the acquisition of products from suppliers for which there exists a formal agreement. [iv]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Preferred Supplier Listing	Medium	Linked Documents
Evaluation Criteria	Medium	Linked Documents
Statements of Work/Agreements	Medium	Linked Documents
Supplier Progress Reports	High	Resource Allocation
Acceptance Test Procedures/Results	High	Test Case/Scenario Status
Discrepancy Reporting	High	Test Case/Scenario Status

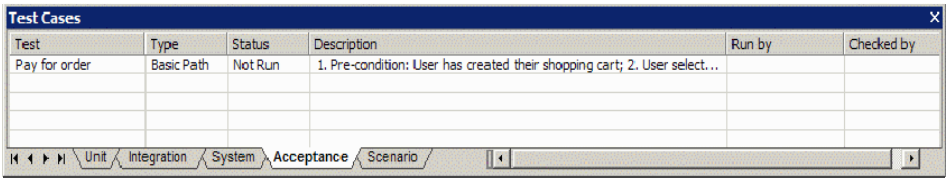

Figure 4 - Test Cases – Established testing criteria is recorded with actual test results and reported defects.

Measurement and Analysis (MA)

Develop and sustain a measurement capability that is used to support management information needs. [v]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Measurement Objectives	Medium	Linked Document
Measurement Specifications	Medium	Linked Document
Data Collection/Storage Procedures	Medium	Linked Document
Data Collection Tool	Medium	SQL and Excel
Analysis Specification/Procedures	Medium	Linked Document
Data Analysis Tools	Low	Rely upon external source
Analysis Results/Reports	Low	Rely upon external source
Stored Data Inventory	Medium	Linked Document

Process and Product Quality Assurance (PPQA)

Provide staff and management with objective insight into processes and associated work products. [vi]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Evaluation Reports	Medium	Linked Documents
Noncompliance Reports	Medium	Linked Documents
Corrective Actions	High	System Issues (Action Items)
Quality Trends	Low	Rely upon external source

Configuration Management (CM)

Establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits. [vii]

CMMI Work Products	Enterprise Architect	EA Implementation
Configuration Item Identification	High	Adaptable to CM system conventions
Configuration Management System	Low	Rely upon third-party product
Change Request Database	High	EA Requirement Element
Configuration Item Baselines	Low	EA Elements only – CI baseline rely upon third-party product
Change Request Tracking	High	Hierarchy Diagram
Revision History of CIs	Low	Rely upon third-party product
Baseline Archives	Low	Rely upon third-party product
Configuration Item Status	Low	Rely upon third-party product
Baseline Delta	Low	EA Elements only – CI baseline rely upon third-party product
Configuration Audits	Low	Link to external source

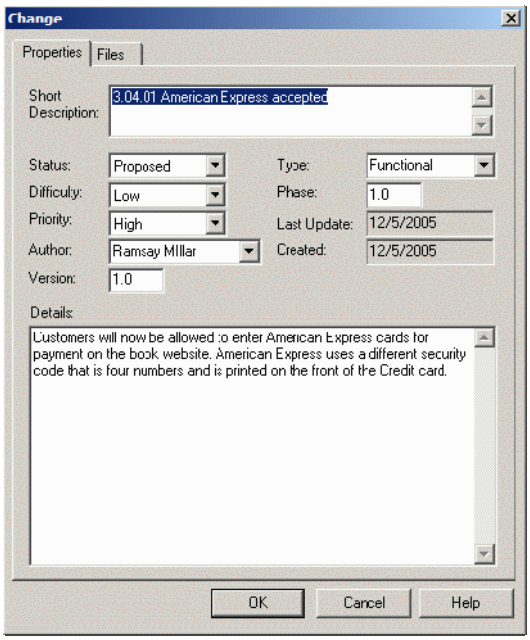

Figure 5 - Change Request – Change requests are documented to include Status, Difficulty, Priority, Type, Phase, supporting details and attached files.

Generic Practices

Process is institutionalized as a managed process. [viii]

CMMI Work Products	Fidelity Level	Enterprise Architect Implementation
Establish an Organizational Policy	Medium	Linked Document
Plan the Process	Medium	Linked Document
Provide Resources	High	Resource Allocation
Assign Responsibility	High	Resource Allocation
Train People	Medium	Linked document
Manage Configurations	Medium	Version Control
Identify & Involve Stakeholders	High	Resource Allocation
Monitor and Control the Process	Low	Rely upon linked sources
Objectively Evaluate Adherence	Low	Rely upon linked sources
Review Status with Management	Medium	Linked Document

Considerations

Cost – Enterprise Architect offers user licenses at a fraction of the costs associated with the other mainstream CASE products. The price per seat is a very attractive feature of this product especially to the small and mid-size corporations.

Training - A corporate cultural change occurs when a team moves from manual methods or legacy UML modeling tools to the discipline of using an agile UML case tool like Enterprise Architect. A formal training program with an emphasis on hands-on activity and real-world exercises is strongly recommended. Although training is necessary for the entire team, we found it essential to have a designated UML custodian that has an in-depth working knowledge of UML to configure work environments and to tailor site specific solutions.

Conclusions

There is no substitute for strong management support and dedicated stakeholder commitment to affectively implement process improvement. Most organizations weigh the benefits and trade-offs of the various methodologies, supporting tools, and consulting services available then develop an approach for CMMI implementation. This paper has focused on a tool-based approach designed to closely satisfy the CMMI specific practices with an emphasis to minimize expenses and maximize reuse capability. Hopefully the information presented here will encourage organizations to engage process improvement with realistic expectations that a tool driven solution may be the optimum choice.

Corporate Profiles

Anteon	iTa Learn Perform Mentor
Anteon is a leading information technology and engineering solutions company providing support to the federal government and international sectors for more than 26 years. Additionally, Anteon is a transition partner with the SEI and has been involved in software process improvement (SPI) initiatives using the capability maturity models (CMM/CMMI) since 1995. **Contacts:** Jack Hunnicutt or David Leonard 700 South Palafox Street, Suite 300 Pensacola, FL 32501 (850) 470-0585	iTa "integrate IT architects" provide consulting and case tool training solutions responding to a growing market need for quality software process improvement. iTa solutions incorporate industry best practices when improving business process, software requirements, analysis, design, development and testing. Our practice focus is on improving quality, reducing risk and increasing the ROI for your software projects. **Contacts:** Ramsay Millar 2336 SE Ocean Blvd Suite 136 Stuart, FL 34996 (877) 249-1252

i-viii Source: (SEI Capability Maturity Model® Integration (CMMI[SM]), Version 1.1)