J2.3 PROCESS MANAGEMENT AND IMPROVEMENT FOR CLASS SYSTEM DEVELOPMENT AND MAINTENANCE

Anita K. Konzak[1]
NOAA/NESDIS, Suitland, Maryland

Richard G. Reynolds[2]
NOAA/NESDIS, Suitland, Maryland

Carlos Martinez[3]
NOAA/NESDIS, Suitland, Maryland

1. INTRODUCTION

The NOAA Comprehensive Large Array-data Stewardship System (CLASS) has implemented a process baseline on multiple development project sites – the NOAA customer in Suitland, MD, and partner development contractors in both Suitland, MD (CLASS-MD), and Fairmont, WV (CLASS-WV) — to achieve the prestigious Software Engineering Institute (SEI) Capability Maturity Model, Integrated (CMMI) rating of Level 2 for the software engineering body of knowledge. The program comprises a total of 72 personnel, consisting of management, development, operations, and support personnel.

Implementation of the process baseline on CLASS has shown:

- Principles in place have shown success
- Repeatable processes and practices work
- Consistent documentation is produced
- Evidence of readiness to the next life cycle phase is prepared and distributed
- Multiple groups/personnel keep informed of the status of the release and release contents
- Basis for measures has been produced, from which improvements can be identified and implemented
- Represents strong project management and control of the project/program
- Represents use of industry standards

Corresponding author addresses:
[1]Anita Konzak, NOAA CLASS, CSC, 5627 Allentown Road, Suite 100, Suitland, MD 20746; e-mail: Anita.Konzak@noaa.gov
[2]Richard G. Reynolds, NOAA OSD, E/OSD-3 NSOF/1210, 4231 Suitland Road, Suitland, Maryland 20746; e-mail: Richard.G.Reynolds@noaa.gov
[3]Carlos Martinez, NOAA CLASS, TMC Technologies, 2110 Pleasant Valley Rd, Fairmont, WV 26554; e-mail: Carlos.Martinez@tmctechnologies.com

2. PROCESS MANAGEMENT OVERVIEW

The policies, processes, procedures, and tools used on the CLASS program were created specifically to support CLASS, but were developed to be adaptable to the process baseline on any size project or program with ease and minimal modification.

The CLASS Team's approach to process baseline management used the model of implementing process in three distinct steps: management, control, and evaluation. This three-step model simplified the approach to understanding the use of a process baseline to support overall management of a project and program.

Management of a project includes the direction of the customer, management activities used on the specific development sites and areas of responsibility, and stakeholder and user involvement. *Control* of a project includes the use of tools, processes, and procedures to manage configured items and the use of integrated teams to manage, govern, and control project-wide activities and impacts. The CLASS project established a Software Engineering Process Group (SEPG) and teams for implementing and integrated management, systems engineering, administration, and operations.

Evaluation on the CLASS project refers to the use of independent quality management to conduct reviews and audits; appointment of development personnel to conduct product reviews and checklists; conduct of lessons-learned sessions to capture activities that work

on the project and improve on processes that are not as effective; and use of project personnel to collect, analyze, and report measurements at monthly and system release cycles. This three-step model approach simplified the understanding of how to use a process baseline to support overall management of a project and program. With all affected organizations understanding this model, management policy and integrated project management, control, and evaluations have been planned and implemented across CLASS.

The process baseline was created and implemented on the CLASS-MD project, receiving a Software Capability Maturity Model (SW-CMM) Level 3 rating in September 2003. The CLASS Team then improved the process baseline through a series of audits and measurement collections and implemented the baseline on the CLASS-WV partner development project. The CLASS-MD Team trained the partner developer, using audits and metrics, to ensure compliance with the established process baseline. The joint CLASS team completed the effort to achieve the SEI CMM-Integration (CMMI) Level 2 rating for the CLASS program in May 2005. The team coordinated and monitored the creation and maintenance of the sets of Process Improvement Indicator Descriptions (PIIDs) for both development sites. PIIDs are databases that contain quality record and other artifacts required to establish and verify compliance with CMMI procedures.

This accomplishment has gained the CLASS project recognition in the NOAA community as a leader in project management and control, software development, stakeholder integration and management, and procurement.

3. INTERGROUP COORDINATION

The CLASS program has identified representative subject matter experts from the CLASS development and operations sites to communicate through regularly scheduled meetings. Each group has created their charter to establish their standard operating procedures. The intergroups established, and a description of their responsibilities, for CLASS includes:

- CLASS Project Management Team – management oversight and coordination
- System Engineering Team – technical oversight and coordination
- Configuration Control Board – change review and control
- CLASS Operations Team – leads operations activities
- System Administration Team – leads system administration activities
- Software Engineering Process Group – process definition and improvement
- Developer Meetings – ensure communication and coordination between developers and development groups

4. PROCESS IMPROVEMENTS

Lessons Learned sessions on CLASS are conducted on a maintenance release cycle (3-4 releases per year) and are held for each development site to ensure that unique development improvements can be attained. CLASS has also implemented lessons-learned sessions that include all development personnel from both sites. This practice strengthens relationships among the groups and maintains excellent communication levels. As part of each lessons-learned session, the CLASS process engineering team has implemented the Six Sigma process-based Failure Modes and Effects Analysis tool to quantitatively identify process improvement opportunities. Working with development management, the process engineering team documents the effect and cause of the lesson, then identifies a prevention to alleviate the negative lesson and a detection to ensure the prevention is effective. These improvements are managed through the CLASS Software Engineering Process Group (SEPG), with measures identified to capture the effectiveness of the improvement opportunity.

5. MEASUREMENTS

Implementation of the process baseline for CLASS has shown results in continual improvement in the products CLASS produces. Examples of measured improvements for CLASS include:

Basis of estimate for configuration change has decreased by 20 percent over past 3 years – In 2002, the average estimate to complete an identified configuration change for CLASS was identified at 25 days per request. Once the process baseline was implemented on CLASS, the quantified measure showed the effort

reduced to 22 days after 1 year, then down to 20 days per change after 2 years. CLASS has measured the effectiveness of their effort against the planned work to continually establish and meet the timeframe to conduct release activities.

*Number of problem reports per configuration change has decreased by 40 percent over past 2 year**s*** – Once the CLASS process baseline was implemented, error rates over a period of 2 years (6 software releases) have improved from .8 errors per configuration change to .5 errors per configuration change. Implementing improvements to CLASS review practices is expected to reduce error rates even further.

*Number of product peer code reviews has increased significantly over past 2 year**s*** – Once the effectiveness of the CLASS process baseline was seen in the form of test readiness reviews, CLASS development personnel realized the benefit of using peer reviews to ensure the quality and consistency of code to conduct configuration changes. CLASS system engineers increased the utilization of code reviews from a negligible measure to a full 75 percent of configuration changes per release. In addition, the system engineering team is reviewing the improvements noticed from the code reviews in implementing increased design reviews for select configuration changes.