The SEL Adapts to Meet Changing Times

Rose S. Pajerski
NASA, Goddard Space Flight Center

Flight Dynamics Division
Greenbelt, MD 20771
(301) 286-3010
rose.pajerski@gsfc.nasa.gov

Victor R. Basili
Computer Science Department/
Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742
(301) 405-2668
basili@cs.umd.edu

Background

Since 1976, the Software Engineering Laboratory (SEL) has been dedicated to understanding and improving the way in which one NASA organization, the Flight Dynamics Division (FDD) at Goddard Space Flight Center, develops, maintains, and manages complex flight dynamics systems. It has done this by developing and refining a continual process improvement approach that allows an organization such as the FDD to fine-tune its process for its particular domain. Experimental software engineering and measurement play a significant role in this approach.

The SEL is a partnership of NASA Goddard, its major software contractor, Computer Sciences Corporation (CSC), and the University of Maryland's (UM) Department of Computer Science. The FDD primarily builds software systems that provide ground-based flight dynamics support for scientific satellites. They fall into two sets: ground systems and simulators. Ground systems are midsize systems that average around 250 thousand source lines of code (KSLOC). Ground system development projects typically last 1 - 2 years. Recent systems have been rehosted to workstations from IBM mainframes, and also contain significant new subsystems written in C and C++. The simulators are smaller systems averaging around 60 KSLOC that provide the test data for the ground systems. Simulator development lasts up to 1 year. Most of the simulators have been built in Ada on workstations. The project characteristics of these systems are shown in Table 1. The SEL is responsible for the management and continual improvement of the software engineering processes used on these FDD projects.

Characteristics	Applications	
	Ground Systems	Simulators
System Size	150 - 400 KSLOC	40 - 80 KSLOC
Project Duration	1.0 – 2.0 years	.5 – 1.0 years
Staffing (technical)	4-10 staff-years	1 – 2 staff-years
Language	C, C++, FORTRAN	Ada, FORTRAN
Hardware	Workstations	Workstations

Table 1. Characteristics of SEL Projects

During the past 20+ years, the SEL's overall goal has remained the same: to improve the FDD's software products and processes in a measured manner. This requires that each development and maintenance effort be viewed, in part, as a SEL experiment, which examines a specific technology or builds a model of interest for use on subsequent efforts. The SEL has undertaken many technology studies while developing operational support systems for numerous NASA spacecraft missions.

The SEL process improvement approach shown in Figure 1 is based on the Quality Improvement Paradigm [Reference 1] in which process changes and new technologies are 1) selected based on a solid *understanding* of organization characteristics, needs, and business goals; 2) piloted and *assessed* using the scientific method to identify those that add value; and 3) *packaged* for broader use throughout the organization. Using this approach, the SEL has successfully established and matured its process improvement program throughout the organization.

The SEL's basic approach toward software process improvement is to first understand and characterize the process and product as they exist to establish a local baseline. Only then can new technologies be introduced and assessed (phase two) with regard to both process changes and product impacts. There are typically several studies ongoing at any one time, which take 1-3 years to complete. The third phase synthesizes the results of the first two phases into various packages such as process tailoring guidance, training materials, and tools and guidebooks. These results are then fed back into the cycle for subsequent projects to use and benefit from.

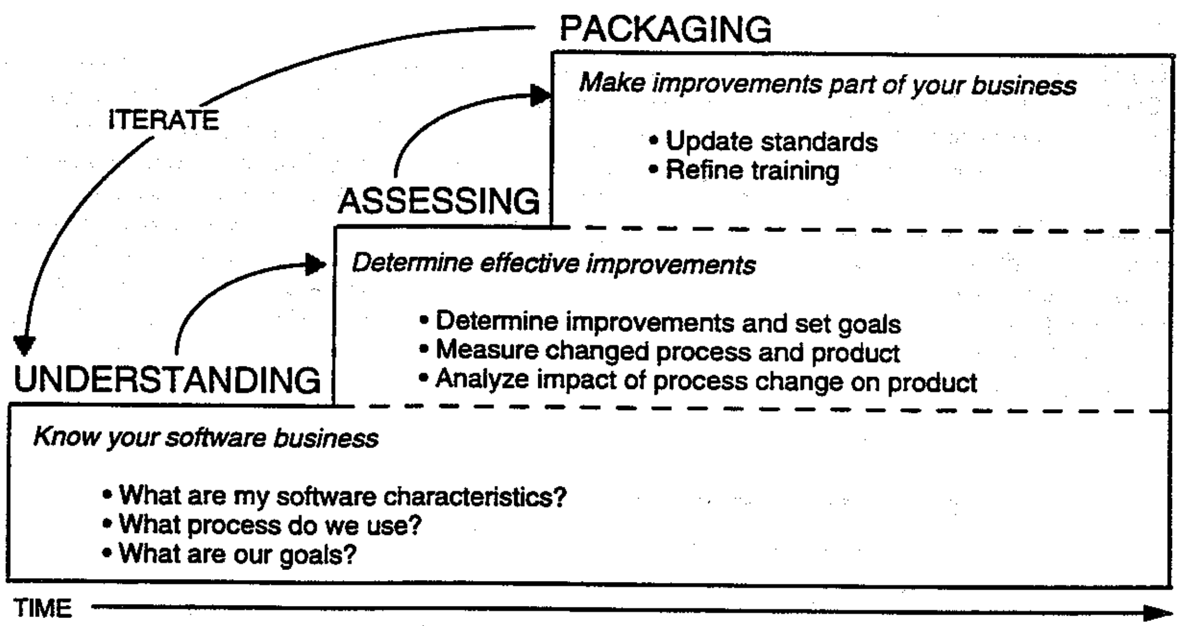

Figure 1. SEL Process Improvement Paradigm

The SEL organization consists of three functional areas: software *developers*, software engineering *process analysts*, and *data base support* (Figure 2). The largest part of the SEL is the 150 to 200 software personnel who are responsible for the development and maintenance of over 4 million source lines of code (SLOC) that provide orbit and attitude ground support for all Goddard missions. Since the SEL was founded, software project personnel have provided software measurement data on over 130 projects. This data has been collected by data base support personnel and stored in the SEL data base for use by software project personnel and process analysts. The process analysts are responsible for defining the experiments and studies, analyzing the data, and producing reports. These reports affect such things as project standards, development procedures, and how projects are managed. The data base support staff is responsible for entering measurement data into the SEL data base, quality assuring the data, and maintaining the data base and its reports.

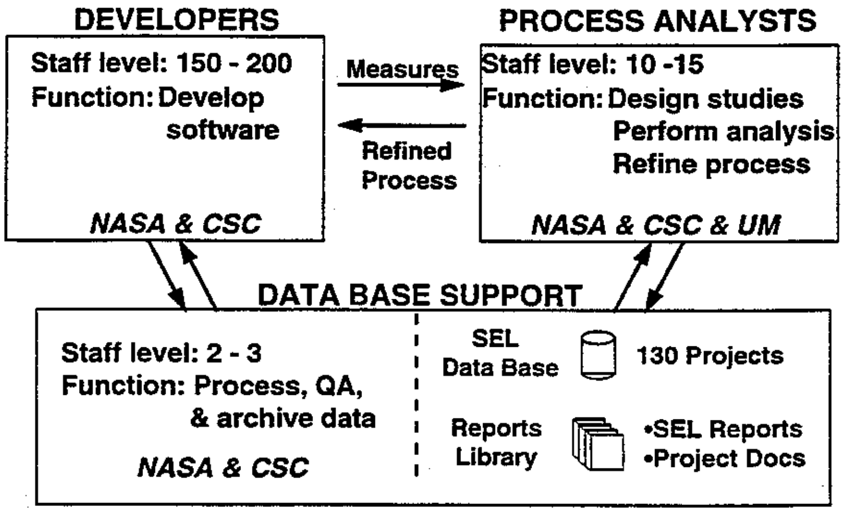

Figure 2. SEL Organizational Structure

Drivers for Change

The SEL has faced a number of changes over the past few years brought about by both environmental and technical factors. These factors include the phase-out of the mainframe systems and subsequent transition to workstations, growth of object-orientated languages (Ada, and C++), and the increasing usage of Commercial-off-the shelf (COTS) scientific application products. However, the latest challenge goes deeper, to a more fundamental, organizational level.

The drivers for this start at the NASA-wide level and extend throughout both Goddard and the local division organizations. The 1997 NASA Strategic Plan has several elements that impact Goddard structures:

- The Enterprise Organizations, such as Mission to Planet Earth and Human Exploration of Space, will become a source of direct funding for SEL studies. This implies a more involved customer who will expect the SEL to be able to show some cost benefit fairly quickly.
- Mission managers will need performance data for both in-house as well as acquired software efforts to show schedule and budget conformance.
- The NASA Software Strategic Plan defines specific goals for software management, assurance, and improvement organizations to attain that may impact SEL activities.

Other drivers arise from the Goddard Strategic Plan and Goddard's new organizational structure. While teams have always provided support for missions, the roles and responsibilities of teams have been expanded. A mission team will now support a specific project throughout all mission phases, thereby involving development organizations earlier in the project. The scope of the development organization has also been broadened to include end-to-end information systems (ground and onboard). The new functionally based Information Systems Center (ISC) shown in Figure 3 will be the focal point for software expertise and a systems support infrastructure throughout Goddard.

Information Systems Center (ISC)

Flight dynamics
Science data processing
Science data visualization and analysis
Command, control, integration and test
Ground and space network
Flight and embedded software
Information system technology

Figure 3. ISC Functions

Given these new strategic and organizational mandates, the SEL has an opportunity to leverage its capabilities to help meet the ISC's expanded responsibilities in several areas:

- Build an improvement organization within the ISC that will increase the competency of its software professionals, thereby increasing the quality of Goddard software systems
- Model and characterize software systems in use on the ground and onboard spacecraft
- Transfer and help tailor proven development and maintenance technologies to new domains, internal and external to GSFC.

Organizational Adaptation

Under the proposed expanded SEL structure, the development organization would expand to include the entire project team while the software engineering process analysts and data base support functions would remain the same (Figure x). However, the scope of work of the process analysts would encompass the end-to-end systems development process, from requirement definition through maintenance and operations. The corresponding metrics used would also change to reflect the additional phases of the system lifecycle under analysis. At a minimum, measures relevant to the requirement definition and operations/delivery processes would need to be included.

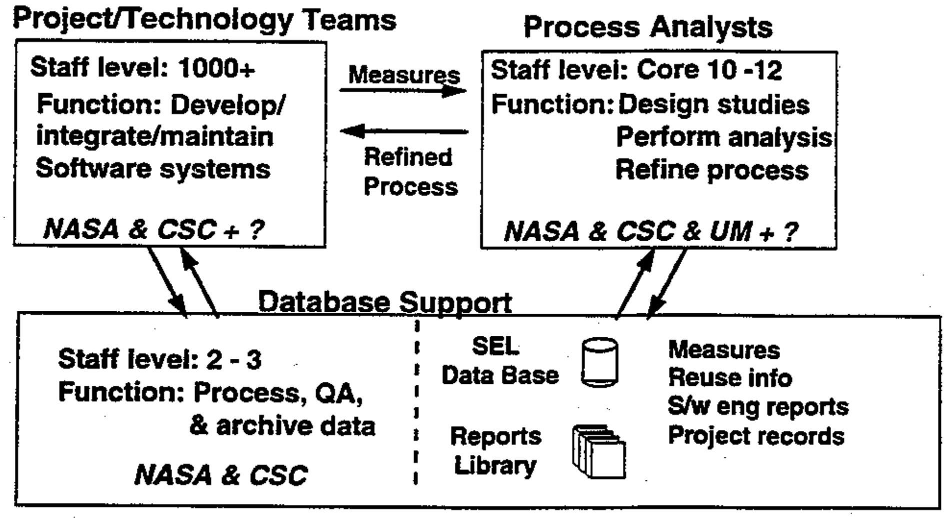

Figure 4. SEL Structure under the ISC

Another organizational issue that will need to be considered is the approach taken in planning SEL work. Over the past several years, the SEL has been managed by a group of 4-6 senior managers who would meet 2-3 times a year to set local improvement goals. Based on these goals, members of the group would propose applied research and study areas for the next 1–2 years. Resource requirements would be discussed and teams formed from the three partner organizations.

With the expansion of the SEL's role in the ISC, planning input for the SEL's activities would be solicited from a broader user community across Goddard including project offices. In their yearly planning, SEL managers would respond to an analysis of these project needs in setting improvement goals, selecting study areas and associated metrics, and feeding back results to all involved. The scope of SEL leadership might also grow to encompass other academic and industry partners; however, this aspect needs further study.

Experimental Adaptation

The SEL has conducted hundreds of process technology studies of different size and duration. Some have been multi-year, multi-application studies (e.g., Cleanroom, Ada) while others have been much smaller and quicker (e.g., testing approach). [References 2,3,4] Over time, this has resulted in refinements to the experimental approach itself in two areas: study selection and approach, and types of analysis performed.

The trend has been to perform smaller studies that build upon one another over time. This has two benefits: quicker feedback to the development groups of useful results, and quicker realization of benefits that then accumulate over time. The independent test team study and the ongoing COTS process study are two examples of this. Figure 5 shows the current approach that emphasizes early deployment of new elements of the process to the development groups as the study proceeds. The impact of these changes on the overall process and product can then be demonstrated and incorporated into the organizational baseline, thereby increasing ISC's competitive position.

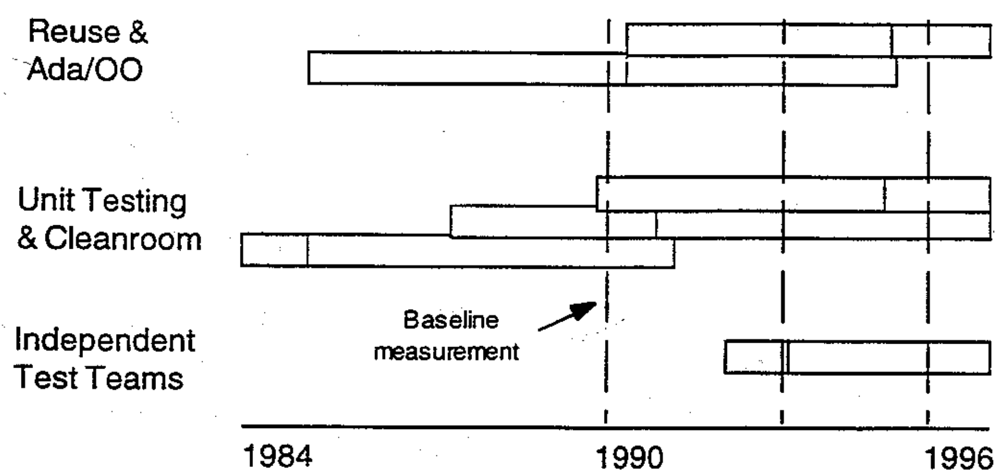

Figure 5. Improvement Cycle Timelines

This emphasis on accelerating analysis has also modified the types of analysis performed. The use of qualitative techniques, such as focused interviews, has increased. The development teams are interviewed at the experiment's start to ensure that their goals and perspectives are factored into the experiment. This domain "discovery " step enhances the SEL process analysts' understanding and facilitates communication throughout the study. Another helpful communication mechanism is the use of online feedback reports to replace face-to-face meetings. As more groups become involved in SEL studies, their use is expected to increase – to become an important technology transfer mechanism also.

Technology Transfer Adaptation

SEL experience with transferring our process technologies and experimental results to the "outside" was predicated on whether the receiving organization was internal to Goddard or external. Internal transfers were supported with a very hands-on approach that included training, tailoring, and impact analysis. However, for external transfers, the users were provided with detailed guides along with some tailoring information but were essentially responsible for implementation and change analysis.

As our understanding of the considerable resources required to be successful at technology transfer has increased, the SEL approach to technology transfer has become more sophisticated. We are developing domain-based techniques to replace the previous "one size fits all" approach. The resulting mechanism considers a range of factors in order to predict the success of a transfer based on key similarities in organization and environment between the organizations involved. The filter can also be used to more easily tailor a particular technology.

Next Steps

Based on the above discussion, there are several steps for the SEL and the ISC to pursue in concert:

1. Profile the ISC organization and establish a new baseline of products developed and processes used in systems development, integration, and maintenance. An understanding of the new operation of the organization is crucial to establish priorities and successfully plan SEL support for them.

2. Select a few projects for focused SEL support – projects that differ in scope (development vs. COTS integration) and organization (in-house vs. contract). This would involve establishing basic measurement mechanisms as well as feedback and reporting procedures for a subset of new or ongoing projects.

3 Evolve the direction of the SEL to include Enterprise representation and new models for leadership. A key ingredient to the SEL's past success has been the close cooperation between development and process analysis groups – and it will be a challenge to replicate this with other groups, perhaps across different companies.

Expanding the scope and support activities of the SEL will not be easy; however, it will position the ISC to be able to improve Goddard's future systems development efforts.

References

[1] Basili, V., "Quantitative Evaluation of a Software Engineering Methodology," Proceedings of the First Pan Pacific Computer Conference, Melborne, Australia, September 1985.

[2] Basili, V. and S. Green, "Software Process Evolution at the SEL," *IEEE Software*, July 1994, pp. 58-66.

[3] Waligora, S., M. Stark, and J. Bailey, "The Impact of Ada and Object-Oriented Design in NASA Goddard's Flight Dynamics Division," *Proceedings of the 13th Annual Washington Ada Symposium (WAdaS96)*, July 1996.

[4] Waligora, S. and R. Coon, "Improving the Software Testing Process in NASA's Software Engineering Laboratory," *Proceedings of the Twentieth Annual Software Engineering Workshop*, Goddard Space Flight Center, December 1995.