N94-36486

PROCESS IMPROVEMENT AS AN INVESTMENT: MEASURING ITS WORTH

19940031979

52-61
12684
p. 21

Frank McGarry
Kellyann Jeletic

SOFTWARE ENGINEERING BRANCH
Code 552
Goddard Space Flight Center
Greenbelt, Maryland 20771
(301) 286-6347
(301) 286-7698

ABSTRACT

This paper discusses return on investment (ROI) generated from software process improvement programs. It details the steps needed to compute ROI and compares these steps from the perspective of two process improvement approaches: the widely known Software Engineering Institute's Capability Maturity Model and the approach employed by the National Aeronautics and Space Administration's (NASA's) Software Engineering Laboratory (SEL). The paper then describes the specific investments made in the SEL over the past 18 years and discusses the improvements gained from this investment by the production organization in the SEL.

INTRODUCTION

For many years, various organizations have put forth significant efforts toward the improvement of software process and product. In recent years, the development of the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) has significantly accelerated interest in the overall improvement process for software. With the development of this model, software development organizations have a relatively clear definition of recommended approaches for attaining better and better levels of software process that, in turn, is expected to result in better and better software products. After six years of experience with the application of the CMM concept, there still is a shortage of empirical evidence quantifying the impact of investments in software process improvement. In general, there has been significant uncertainty in the return on investment stemming from process improvement activities. As organizations invest resources in software process improvement efforts, they need to understand what they are getting for their money and determine whether there has been any benefit from this investment.

This paper details the steps needed to compute return on investment (ROI) and compares these steps from the perspective of two process improvement approaches: the widely known Software Engineering Institute's

Capability Maturity Model and the approach employed by the National Aeronautics and Space Administration's (NASA) Software Engineering Laboratory (SEL). It then describes the specific investments made by the SEL over the past 18 years and discusses the benefits gained from this investment by this production organization.

SEL OVERVIEW

The SEL is an organization sponsored by NASA's Goddard Space Flight Center (GSFC) which was created to investigate the effectiveness of software engineering technologies when applied to the development of applications software. The SEL was established in 1976 and has three primary organizational members: NASA/GSFC's Software Engineering Branch, the University of Maryland's Department of Computer Science, and the Computer Sciences Corporation's (CSC's) Software Engineering Operation. The goals of the SEL are (1) to understand the software development process in the GSFC environment; (2) to measure the effect of various methodologies, tools, and models on this process; and (3) to identify and then apply successful development practices.

Within the SEL, over 100 production projects have been monitored and studied over an 18 year period to assess the impact that process change has on the developed software products. These production projects result in software that is used for ground support for GSFC missions and is typically used to carry out all Flight Dynamics functions at the GSFC. These software projects range in size from 4 or 5 thousand (K) source lines of code (SLOC) to over 1 million SLOC, with a typical size of 100-300 KSLOC.

In carrying out these 100 'experiments' with software process, the SEL has accumulated detailed information on specific processes used for each project as well as the resultant product characteristics such as cost, error rates, cycle time, rework required, etc. With this information, some insight can be gained into the ROI that is attained with the usage of particular process changes within the environment.

CHARACTERISTICS OF TWO SOFTWARE PROCESS IMPROVEMENT PARADIGMS (CMM AND SEL)

Although the paradigm used by the SEL differs from the SEI's Capability Maturity Model (CMM), both approaches share the underlying principle of continuous, sustained software process improvement. The CMM focuses on improving an organization's software process by evolving through a series of maturity levels to attain the ultimate goal, becoming a continuously improving organization ('level 5'). At each level, the organization must meet a set of well-defined criteria to advance to the next level or beyond.

Within the CMM, an organization strives to mature to a continuously improving process. To do so, the organization must advance through the following maturity levels [Reference 1] where the organization's software process is defined as:

- Level 1 - an ad hoc process
- Level 2 - a repeatable and more disciplined process
- Level 3 - a standard, consistent, and defined process
- Level 4 - a predictable and manageable process
- Level 5 - an optimizing and continuously improving process

The SEL's process improvement paradigm consists of a three step iterative process driven by the specific goals of an organization (e.g., to decrease average error rates) and the experience gained from earlier development efforts (e.g., most errors are interface errors). These three steps include:

1) *Understanding* - a baseline of an organization's software process and product is developed. How is the organization's software business done? What is the lifecycle process? What standards are used? What are the characteristics of its software

product (e.g., cost, error rates, productivity)?

2) *Assessing* - based on the goals of an organization (e.g., reduce error rates), some change is introduced to the process and the subsequent result of that change is assessed.

3) *Packaging* - once improvements have been identified and verified, they are packaged in some tangible form (e.g., training, standards) and infused back into the organization's process.

These three steps are performed iteratively and continuously over time.

The two process improvement paradigms, the CMM and SEL, are depicted in Figures 1 and 2, respectively.

HOW IS ROI COMPUTED?

With any process improvement approach, an organization is eager to determine what it has gained from its investment. There are five steps necessary to determine the benefits gained from investing in software process improvement. These are:

(1) Define goals. The organization must set goals for what is to be improved.

(2) Produce a baseline. The organization must establish a basic understanding of its current software process and product.

(3) Invest in change. To improve anything, change must first be made. An investment in this change must be made.

(4) Assess change. Once a change has been made, its effects must be measured to determine if any improvement has been achieved.

(5) Measure ROI. Has the investment in process improvement been a success? What has the investment been and what has been gained from this investment? The ROI must be measured by (a) determining what resources have been expended for software process improvement, (b) establishing what improvements, both quantitative and qualitative, have been achieved, and (c) determining the difference between the investment made and the benefits obtained. Has the investment been worthwhile?

How are these steps achieved within the framework of the CMM and SEL process improvement approaches? Each step is addressed below from the perspective of both approaches.

Throughout this paper, 'process' refers to the characteristics of how an organization develops and maintains software. 'Process' includes the organization's tools, standards, policies, life cycle, management approaches, etc. It also includes all measures reflecting these items such as effort distribution, error distribution, profile of software change and growth rate, etc. 'Product' refers to the characteristics of the resultant software including productivity, reuse levels, error rates, cycle time to produce, etc.

1 - DEFINING GOALS:

Each organization must set goals for what is to be improved. With the CMM, the goal is generalized, i.e., to improve the software process. With the SEL, goals are product-driven and vary from organization to organization.

CMM: There is a generalized, domain-independent goal that focuses on process. Every organization strives to improve the software process and, ultimately, evolve to a continuously improving, optimizing process (maturity level 5). Organizations A and B both try to improve their processes and become level 5 organizations, thereby minimizing any risk incurred because of software development.

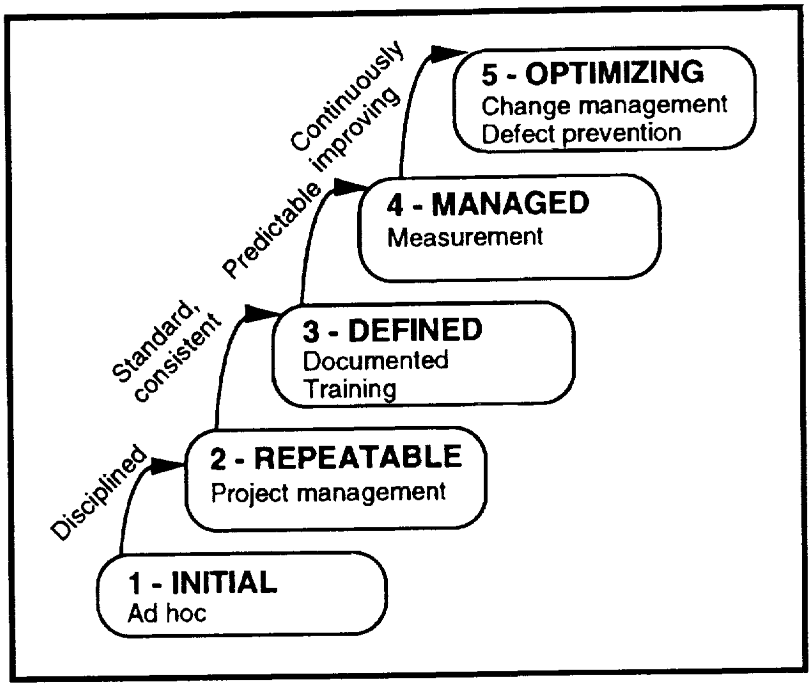


Figure 1. CMM Process Improvement Paradigm

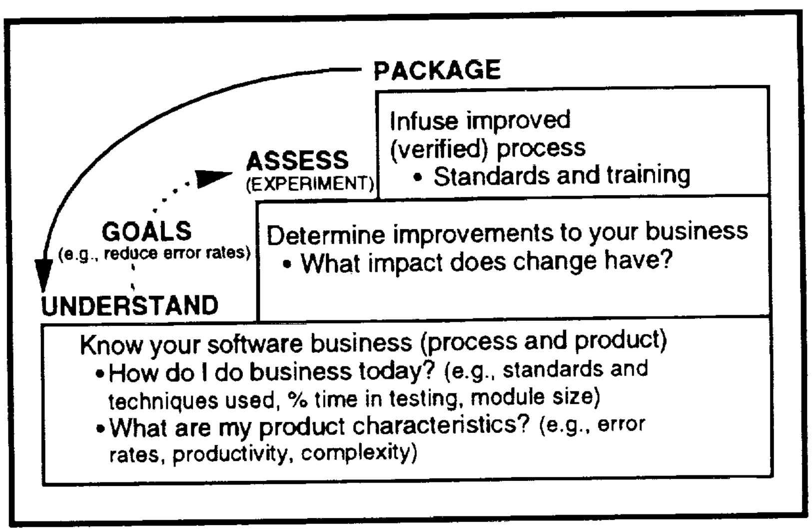


Figure 2. SEL Process Improvement Paradigm

SEL: The emphasis of the SEL approach is to improve the organization's software product. Goals vary from organization to organization and are driven by product, not process characteristics. Organization A may attempt to improve reliability by decreasing error rates. Organization B may strive to decrease development cycle time. Goals are domain-dependent.

2 - PRODUCE BASELINE

Each organization must establish a basic understanding (baseline) of its current software process and product. The CMM baseline is process-based and established against a 'common yardstick.' The SEL baseline is domain-dependent and is both process- and product-based.

CMM: Baselining within the CMM is achieved by performing an assessment of the organization's process. This assessment is made against well-established criteria defined by the SEI [Reference 1] and the organization is baselined at some maturity level. These criteria enable comparisons across domains since every organization is assessed against the same criteria, a 'common yardstick.' The same elements are examined for every organization: does it have good standards, what is its training program like, how is its measurement program, etc. Based on the examination of these criteria, the organization is baselined at some maturity level.

SEL: Baselining involves understanding the process and product of each individual organization. This baseline is organization-dependent (or domain-dependent). Unlike the CMM, there is no common yardstick enabling comparison across organizations. Some factors need to be characterized (baselined) by all organizations, e.g., how much software exists, what process is followed, what standards are used, what is the distribution of effort across lifecycle phases, etc. Other factors of interest depend on the goals of the organization. Organization A, for example, would want to baseline its error rate, while Organization B needs to determine its development cycle time.

The SEL process improvement approach emphasizes introducing change to attain process improvement. The effects of changes to process can only be measured by comparing them to the existing baseline. Understanding is a critical and continually needed element of the SEL approach.

Figures 3 and 4 are examples of the SEL's baseline measures. They represent data from Flight Dynamics projects as specified on the individual figures. Figure 3 depicts baseline values pertaining to process. It shows the SEL's typical effort distribution and classes of error. Figure 4 depicts baseline values associated with product. It shows the SEL's typical error rates, cost, and level of code reuse.

These examples represent some elements that may be characterized by an organization baselining its process and product.

3 - INVEST IN CHANGE

Organizations striving for software process improvement must invest in change. Within the CMM, the common yardstick drives change. Within the SEL, organizational goals and experiences drive change.

CMM: The CMM's common yardstick drives change. That is, the elements by which the CMM assesses maturity levels drive change. If an organization is baselined at some level, it will change elements necessary to get to the next maturity level. If an improved measurement program is needed to advance to another maturity level, the organization will focus on changing its measurement program to meet the CMM criteria. This common yardstick enables a common roadmap to success -- continuous improvement.

SEL: The goals and experiences of individual organizations drive changes. Changes to the process are made in an attempt to improve the product. An organization interested in increasing its level of reuse will invest in changes that focus on that improvement goal. For instance, they might decide to experiment

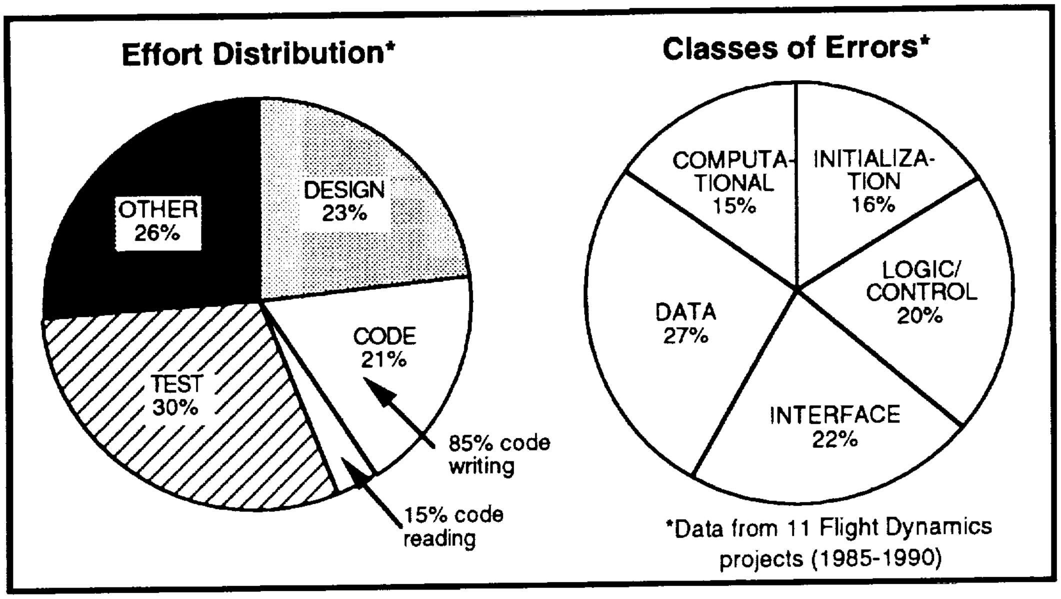


Figure 3. Sample SEL Process Baseline

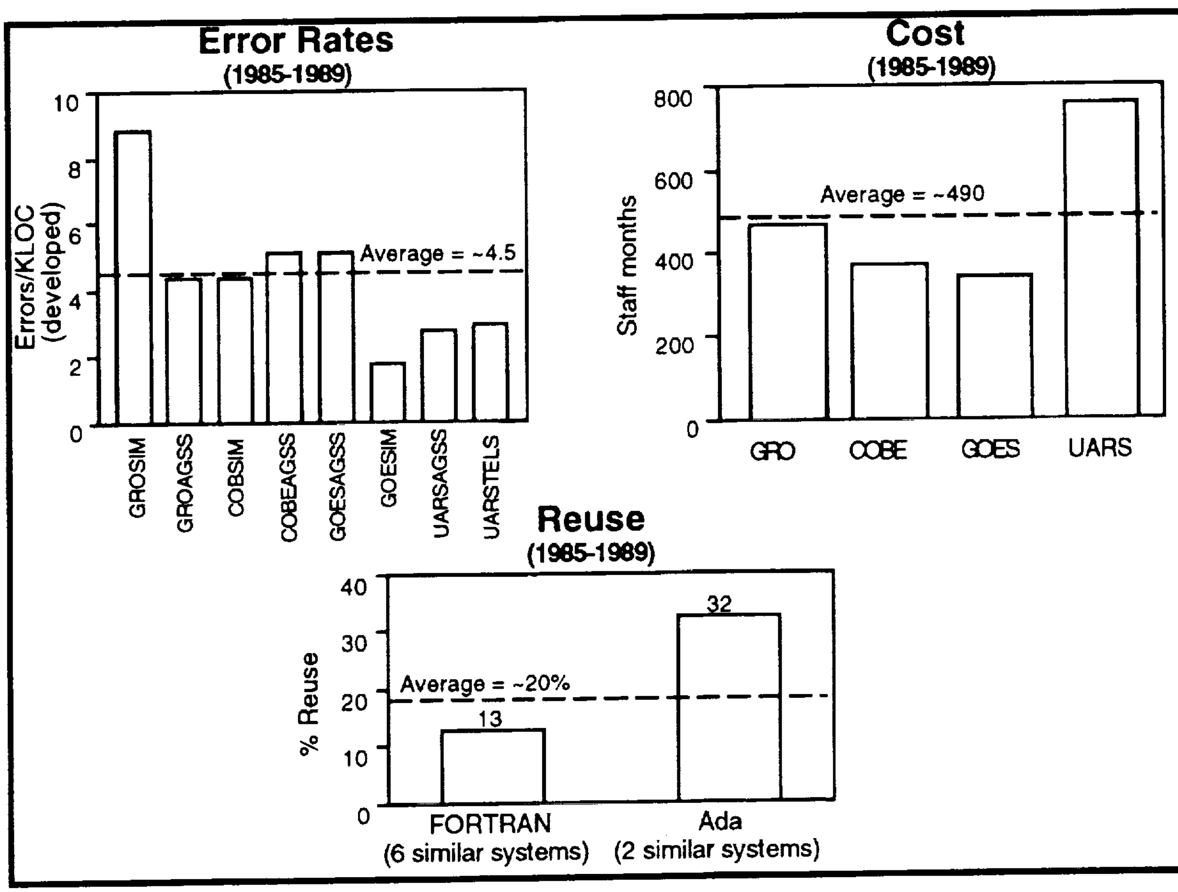


Figure 4. Sample SEL Product Baseline

with object-oriented design (OOD) to improve reuse. The organization interested in reducing error rates might decide to experiment with the Cleanroom methodology [Reference 2]. Each organization (or domain) must identify the most appropriate process changes to achieve its product goals.

The CMM is an excellent model of potential process changes that could be selected. Various elements of the model (e.g., key process areas (KPAs)) have emphasis on specific product improvements that can help in selecting potential changes in the SEL model.

4 - ASSESS CHANGE

Each organization must introduce change to make some improvement. An assessment of the changes must be made to determine if there has been improvement. The CMM assesses change by reassessing the process. The SEL assessment of change is domain-dependent and focuses on both process and product.

CMM: With the CMM, assessment of change is accomplished by reassessing the process. An organization is baselined at one level, makes changes to try to attain a higher level, and is then reassessed to determine if it has progressed to another level. Success is measured by process change. The ultimate success is changing the process until it is a continuously improving process. The organization achieves the highest maturity level rating, that is, advancing to level 5. The measure of success is domain-independent, since all organizations are measured against the same criteria, a common yardstick.

SEL: Assessment of change is domain-dependent. An improvement goal is set, change to the process made, change to the process and product examined and verified, and the effect of change evaluated against the original goal. Success is measured by product change and is determined based on the goals of the individual organization. The organization attempting to improve its reliability would institute a change, e.g., the Cleanroom methodology, to try to reduce its error rates. It would then assess the result of the Cleanroom experiment based on its original goals. What were the baseline error rates? What were the error rates resulting from the Cleanroom experiment? Did Cleanroom reduce error rates? The organization attempting to attain higher levels of reuse would make a change, e.g., OOD. Similarly, it needs to determine the level of reuse achieved using OOD and compare these reuse levels with the original baseline. The SEL examines both changes to process data and changes to product data.

Figures 5 and 6 show some sample assessments from the SEL representing process and product data. They represent data from actual Flight Dynamics projects as specified on the individual figures. Figure 5 depicts a process assessment showing the impact of a technology (Cleanroom) on the SEL's baseline effort distribution. Figure 6 shows an assessment of SEL products for the period 1990-1993. The error rates, cost, and level of reuse are reexamined to determine if there was any change from the early baseline (1985-1989) shown previously in Figure 4.

These examples also reemphasize the need for baselining of both process and product. Without the basic understanding provided by the baseline, no change can be assessed.

5 - MEASURE ROI

Goals have been set. Baselines have been established. Investment in change has been made. Changes have been introduced and their effect assessed based on the original goals and the baseline values. Organizations must now determine if the results of change have been successful. Once 'success' has been determined, then they can attempt to answer the question, "Has the investment been worth it?"

CMM: The CMM measure of success is domain-independent and is the same as its generalized goal. An organization is successful if its process becomes mature and it becomes a continuously improving,

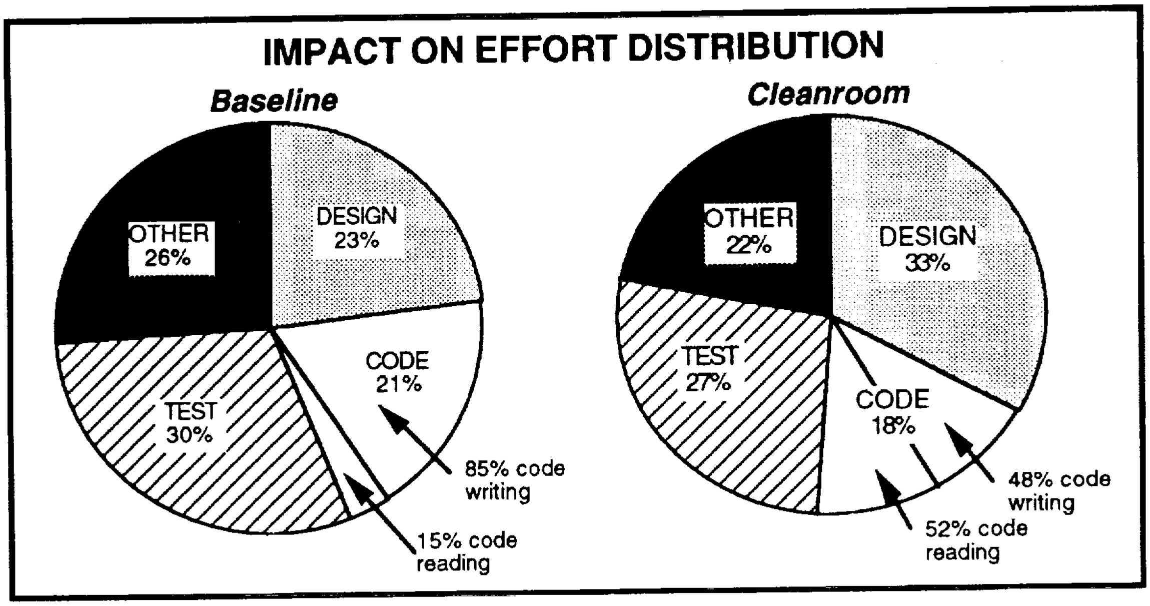


Figure 5. Sample SEL <u>Process</u> Assessment

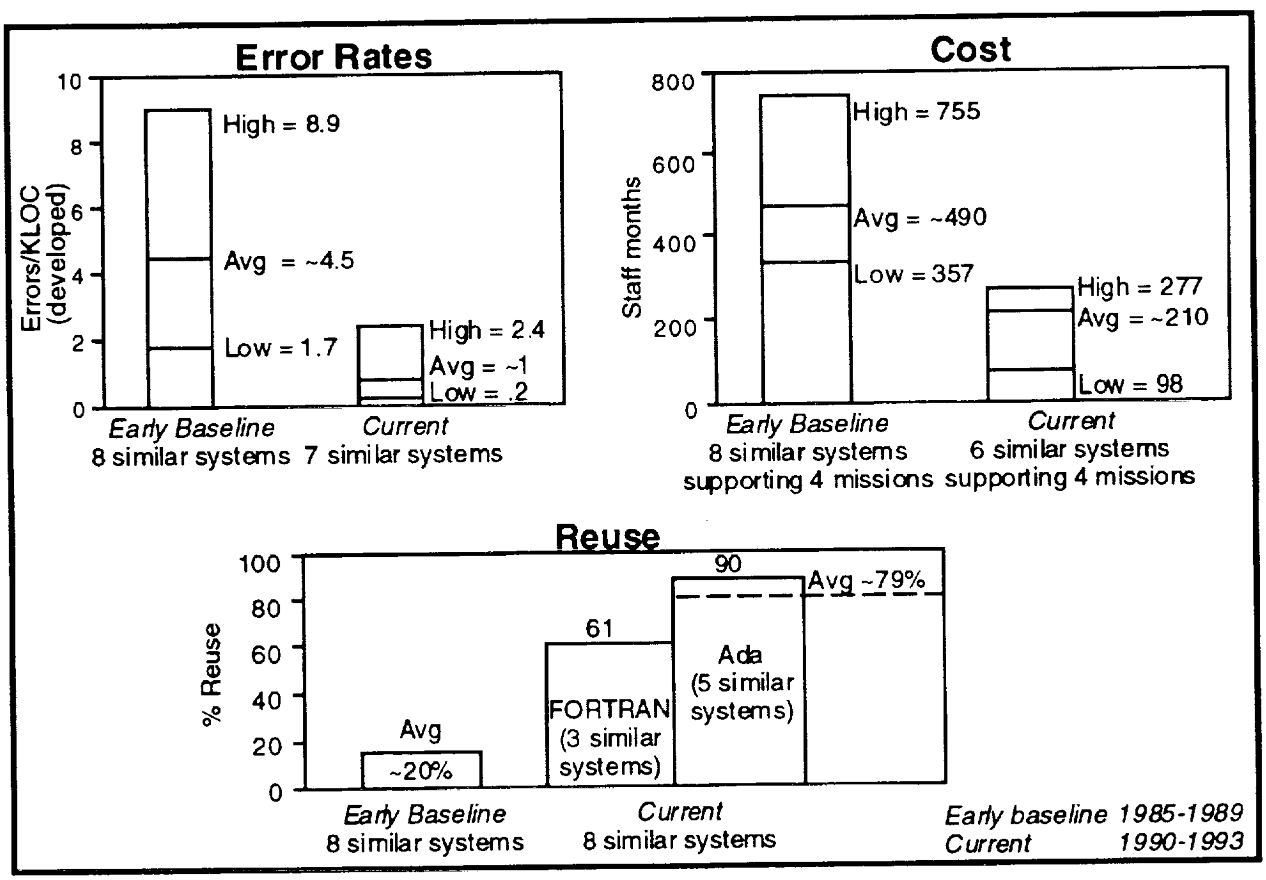


Figure 6. Sample SEL <u>Product</u> Assessment

optimizing process, a maturity level 5. Organizations progress to higher levels and, in doing so, expect to reduce risk and generate better products. Success can easily be determined, but how is ROI determined? After six years of experience with the application of the CMM, there is still no clear, accepted mechanism for determining the value of return for the investment required to implement software process improvement programs.

SEL: The SEL measure of success will vary from organization to organization. Success depends on the goals set by the individual organization. Success for Organization A is decreased error rates; success for Organization B is decreased cycle time. How is ROI determined? The baseline of both process and product and the assessment of change to the baseline result in quantified measures that can be examined to determine the ROI in process improvement. The remainder of this section discusses the ROI for the SEL's process improvement paradigm.

THE ROI FOR THE SEL

GSFC's Flight Dynamics Division (FDD) is the production organization with which the SEL is associated. FDD software development is driven and guided by the SEL process improvement paradigm. Over the past 18 years, the FDD has invested approximately 210 million dollars ($M) in software development activities. Of this amount, the FDD has invested approximately 11%, roughly $24M, in software process improvement. Figure 7 shows the breakdown of this investment. About 1.5% of the total investment (<$3M) is attributed to project overhead including form completion and collection, training, and other similar activities. Another 3% ($6M) was spent processing data: archiving data, maintaining the data base, quality assuring the data, etc. The largest part of this investment has been in analyzing the data. About 7% ($15M) has been spent defining experiments, analyzing results of SEL experiments, developing the SEL models and processes, producing software policies and standards (e.g., References 3 and 4), developing training material, carrying out training in changing processes, and other activities associated with improving the FDD's software products and process.

Has the FDD's 11% investment in software process improvement been worth it? In comparing projects developed in the mid 1980s to those developed in the early 1990s, several significant benefits have been achieved. Figure 6 depicted some of these results. The average level of reuse has increased by 300%, from ~20% to nearly 79% for similar classes of software. Reliability (errors/KSLOC) has improved significantly as the error rate has decreased by 75% from 4.5 to 1 error/KSLOC. The cost of developing Flight Dynamics software has also decreased significantly. The average cost of software per mission has decreased by 55% from ~490 staff months (SM) to ~210 SM.

These quantifiable improvements are complemented by more subjective ones. The SEL's process improvement activities have resulted in many impacts to the software production organization. First, the SEL integrates and focuses activities that were previously disparate. Training, standards, policies, technology insertion, and measurement have gradually become integrated as a result of the SEL's process improvement approach. Figure 8 depicts these items with respect to the three steps of the SEL improvement approach.

Second, there has been a cultural change within the production organization. The developers have become an integral part of process change. In fact, their experiences are the basis for process change. Developers have become more intimately involved with the SEL process improvement approach. For instance, developers on early Cleanroom experiments drove the development of a Cleanroom process handbook for use on later Cleanroom projects. By doing so, they packaged their experiences for future use. Another cultural change lies with the software being produced. Software development within the FDD is now process-driven and much less people-driven. The process is so

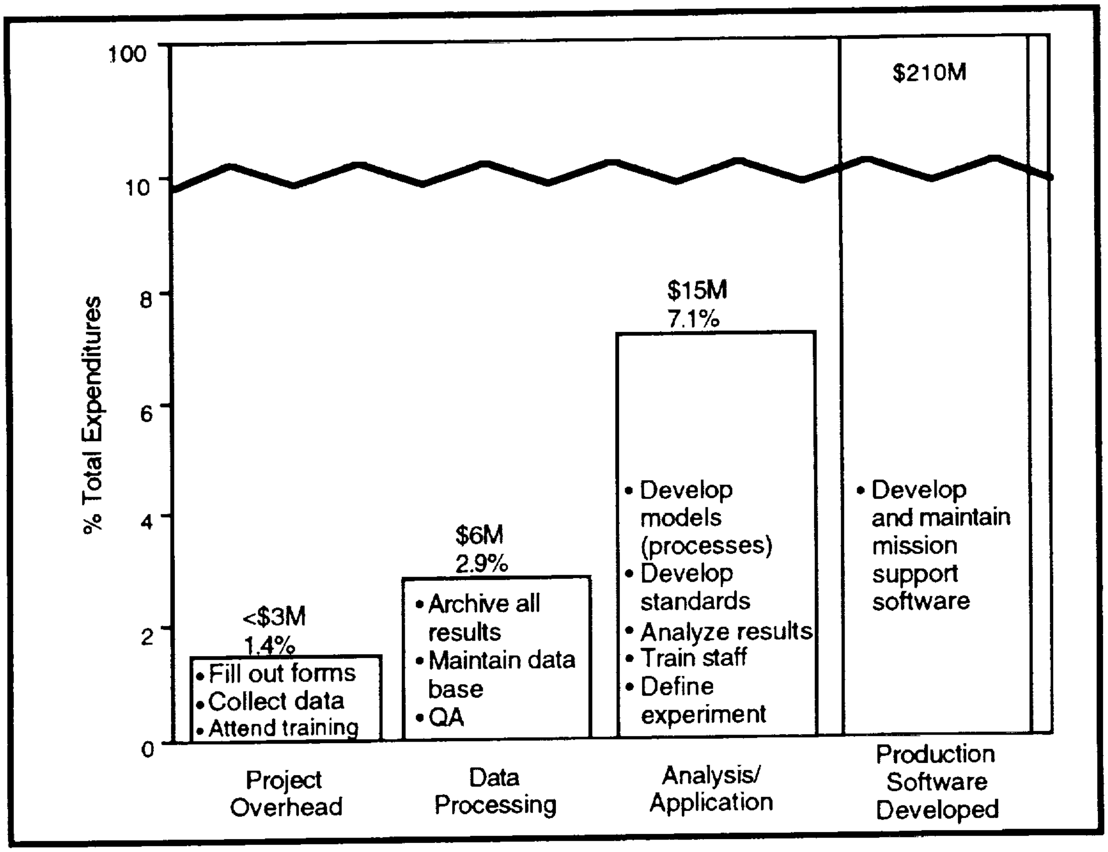


Figure 7. FDD Investment in the SEL and Software Process Improvement

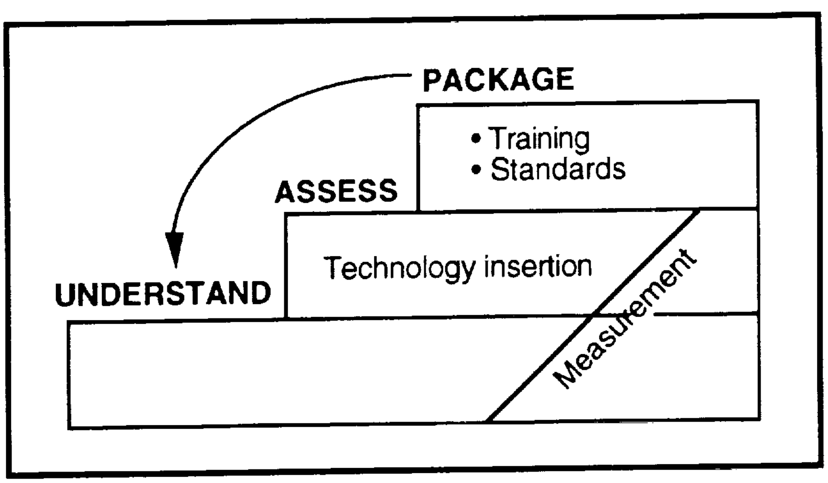


Figure 8. Integration of Software Activities via the SEL Process Improvement Paradigm

well-defined that 'heroes' are not necessary for the well being of a software project.

Finally, there is now a focused role of software engineering research. This research has become goal and product-driven rather than being performed in an ad hoc fashion. There is also a well-established mechanism by which experimentation, assessment, and adoption of technologies are performed. Within the SEL, process improvement has driven organizational evolution and optimized the allocation of software-related resources. While not quantifiable, these have been significant benefits achieved from the 11% investment in software process improvement.

Although there has been significant improvement in the software products in the SEL, there is no way of determining how much of this improvement is attributable to software process improvements and how much is attributable to normal improvements in technology. There have been significant changes to technology such as available tools, support environments, better operating systems, better trained personnel, work environments, faster machines, etc., but there has been no attempt by the SEL to distinguish between improvements driven by the technology maturation vs. software process maturation.

SUMMARY

As already discussed, there have been substantial benefits gained from the investment made in the SEL process improvement activities in the areas of level of reuse, reliability, and cost. While these benefits were being attained, the software being produced was also increasing in complexity (Figure 9). As a result of the SEL, the FDD was able to produce more complex software with more functionality while improving reliability and reducing cost.

The FDD's $24M investment over the past two decades has resulted in substantial benefits for the Division itself and many other organizations. As NASA focuses more on technology transfer, the latter may become a more significant factor in evaluating the ROI for the SEL. Not only has the SEL improved the software process and products of its own production organization, GSFC's Flight Dynamics Division, but it has shared these experiences with many other software organizations both within and outside the Agency. The impact on other organizations cannot be measured, but it certainly is a factor to be considered when determining the value added by the SEL and its process improvement paradigm.

REFERENCES

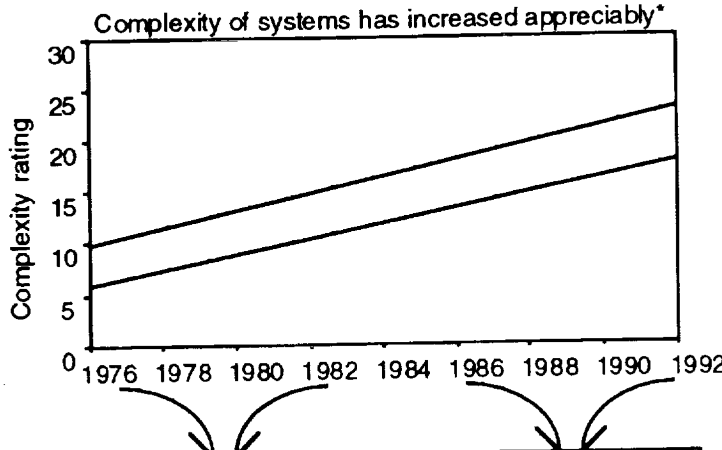
1. Paulk, M., B. Curtis, M. Chrissis, and C. Weber. *Capability Maturity Model for Software, Version 1.1,* Software Engineering Institute, Carnegie Mellon University, CMU/SEI-93-TR-24, February 1993.

2. *Software Engineering Laboratory (SEL) Cleanroom Process Model,* S. Green, NASA Goddard Space Flight Center, Software Engineering Laboratory, SEL-91-004, November 1991.

3. *Manager's Handbook for Software Development (Revision 1),* L. Landis, F. McGarry, R. Pajerski, S. Waligora, et al. NASA Goddard Space Flight Center, Software Engineering Laboratory, SEL-84-101, November 1990.

4. *Recommended Approach to Software Development (Revision 3),* L. Landis, F. McGarry, R. Pajerski, S. Waligora, et al. NASA Goddard Space Flight Center, Software Engineering Laboratory, SEL-81-305, June 1992.

5. Boland, D., *A Study on Size and Reuse Trends in Attitude Ground Support Systems (AGSSs) Developed for the Flight Dynamics Division (FDD) (1976-1988),* Computer Sciences Corporation, CSC/TM-89/6031, February 1989.



	Late '70s-Early '80s	Late '80s-Early '90s
Control	Spin stabilized	3-axis stabilized
Sensors	1	8 to 11
Torquers	1	2 to 3
OBC	Analog simple control	Digital autonomous control
Telemetry	5	12 to 15
Data Rates	2.2 kb/s	32 kb/s
Accuracy	1 degree	0.02 degree

*[Reference 5]

Figure 9. FDD Software Complexity

PROCESS IMPROVEMENT AS AN INVESTMENT: MEASURING ITS WORTH

Frank E. McGarry NASA/GSFC
Kellyann F. Jeletic NASA/GSFC

A846 001

MEASURING ROI* FOR PROCESS IMPROVEMENT

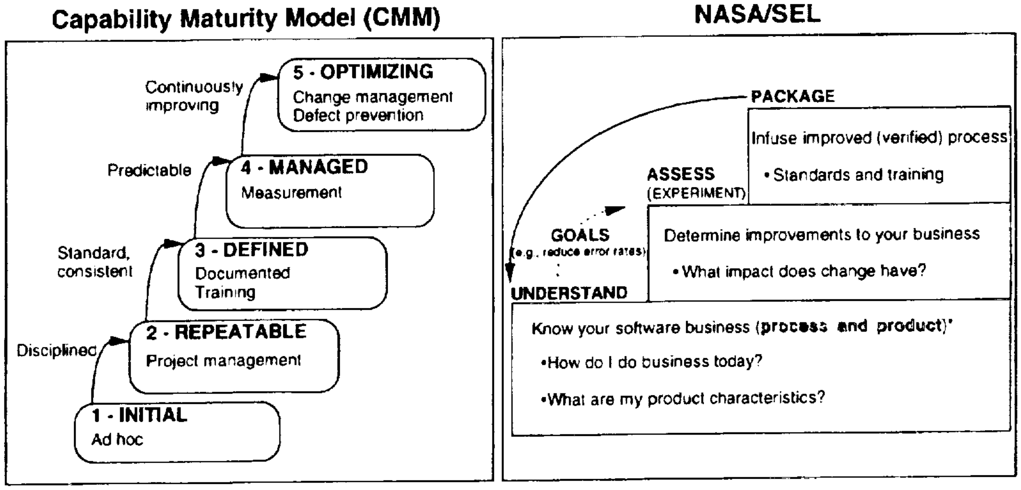
TOPICS OF DISCUSSION

1. CHARACTERISTICS OF TWO PROCESS IMPROVEMENT PARADIGMS
2. INFORMATION NEEDED TO DETERMINE ROI
3. MEASURING ROI IN NASA/SEL

*ROI = RETURN ON INVESTMENT

A846 002

TWO PARADIGMS FOR PROCESS IMPROVEMENT



* Process:
 • How do you do business (e.g., standards and techniques used)
 • Associated measurements (e.g., % time in testing, module size)
Product:
 • End item attributes (e.g., error rates, productivity, complexity)

93M11U1.003

TWO PARADIGM FOR PROCESS IMPROVEMENT

STEPS	CMM EMPHASIS	NASA/SEL EMPHASIS
1. DEFINE GOALS	IMPROVE PROCESS (GET TO HIGHER LEVEL)	IMPROVE PRODUCT
2. PRODUCE BASELINE	PROCESS "ASSESSMENT" (AGAINST ONE YARDSTICK)	PROCESS AND PRODUCT UNDERSTANDING
3. INVEST IN CHANGE	COMMON YARDSTICK DRIVES CHANGE	EXPERIENCES AND GOALS DRIVE CHANGE
4. ASSESS CHANGE	REASSESS PROCESS SUCCESS - HIGHER LEVEL	REEXAMINE PROCESS AND PRODUCT SUCCESS - BETTER PRODUCT
5. MEASURE ROI		TODAY'S TOPIC

A846.006

SOFTWARE PROCESS IMPROVEMENT

STEP 1 - DEFINE GOALS

	CMM PARADIGM	NASA/SEL PARADIGM
	EMPHASIS -- IMPROVE PROCESS	EMPHASIS -- IMPROVE PRODUCT
ORGANIZATION 1	"GET TO LEVEL 5"	INCREASE **LEVEL OF REUSE**
ORGANIZATION 2	"GET TO LEVEL 5"	DECREASE **ERROR RATES**

SEL: IMPROVEMENT GOALS MAY VARY ACROSS DOMAINS

A846 007

SOFTWARE PROCESS IMPROVEMENT

STEP 2 - PRODUCE BASELINE

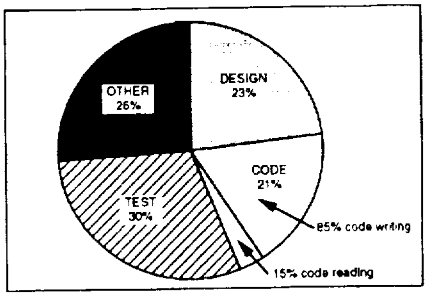
	CMM PARADIGM		NASA/SEL PARADIGM
	EMPHASIS -- PROCESS ASSESSMENT		EMPHASIS -- PROCESS AND PRODUCT UNDERSTANDING
ORG 1	STANDARDS - GOOD TRAINING - WEAK MEASUREMENT - WEAK	➡ LEVEL 1	WHAT IS YOUR DOMAIN? WHAT STANDARDS DO YOU USE? WHAT IS YOUR **LEVEL OF REUSE?**
ORG 2	STANDARDS - WEAK TRAINING - GOOD MEASUREMENT - GOOD	➡ LEVEL 3	WHAT IS YOUR DOMAIN? WHAT STANDARDS DO YOU USE? WHAT IS YOUR **ERROR RATE?**

SEL: MEASURES ARE DOMAIN-DEPENDENT
NO COMPARATIVE MEASURE ACROSS DOMAINS
CMM: HAS COMMON "YARDSTICK", CAN COMPARE ACROSS DOMAINS

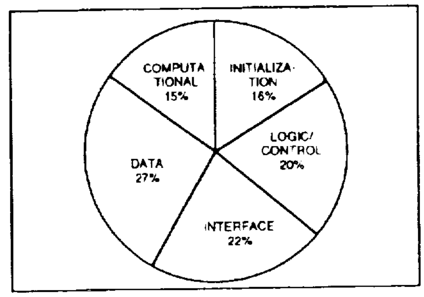
A846 008

NASA/SEL PROCESS BASELINE EXAMPLE

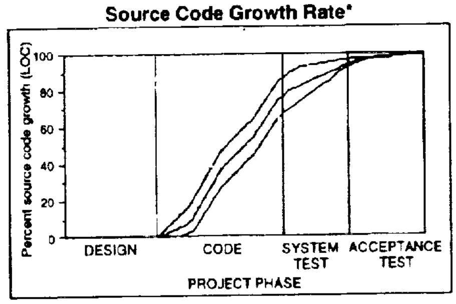
Effort Distribution*



Classes of Errors*



Source Code Growth Rate*

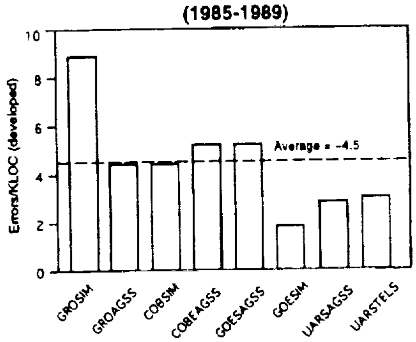


*Data from 11 Flight Dynamics projects (mid 1980s)

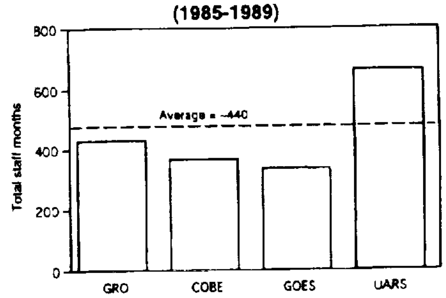
93M11U1.007

NASA/SEL PRODUCT BASELINE EXAMPLE

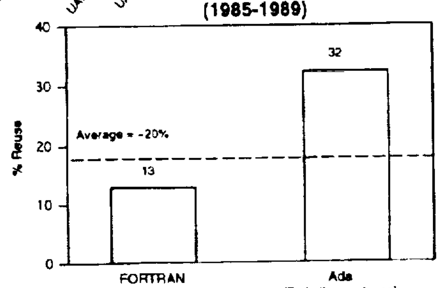
Error Rates (1985-1989)



Cost (1985-1989)



Reuse (1985-1989)



93M11U1.008

SOFTWARE PROCESS IMPROVEMENT

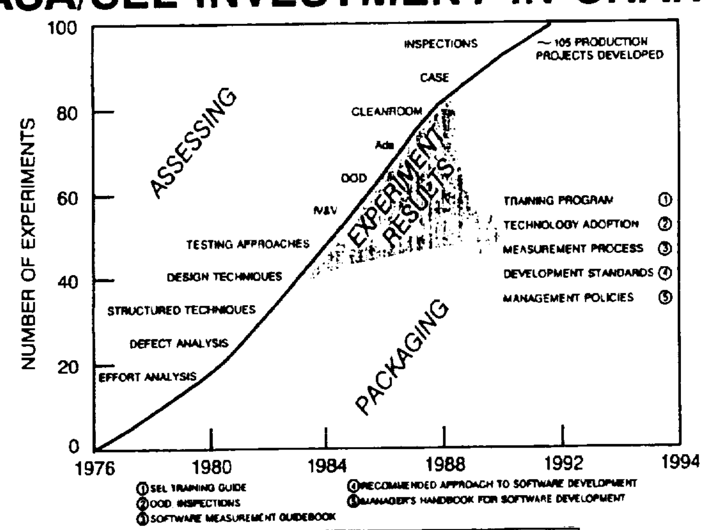
STEP 3 - INVEST IN CHANGE

	CMM PARADIGM	NASA/SEL PARADIGM
	EMPHASIS -- COMMON YARDSTICK DRIVES CHANGE	EMPHASIS -- EXPERIENCES AND GOALS DRIVE CHANGE
ORGANIZATION 1	MAINTAIN GOOD STANDARDS IMPROVE TRAINING IMPROVE MEASUREMENT	EXPERIMENT WITH OOD TO IMPROVE **REUSE** - TRAIN IN OOD
ORGANIZATION 2	WRITE BETTER STANDARDS MAINTAIN GOOD TRAINING MAINTAIN GOOD MEASUREMENT	EXPERIMENT WITH CLEANROOM FOR LOWER **ERROR RATES** - DEVELOP CLEANROOM PROCESS HANDBOOK

SEL: EACH DOMAIN MUST IDENTIFY MOST APPROPRIATE PROCESS CHANGE

A846.011

SOFTWARE PROCESS IMPROVEMENT NASA/SEL INVESTMENT IN CHANGE



EXPERIENCE DRIVES PROCESS CHANGE

A846.012

SOFTWARE PROCESS IMPROVEMENT

STEP 4 - ASSESS CHANGE

	CMM PARADIGM		NASA/SEL PARADIGM
	EMPHASIS -- REASSESS PROCESS		EMPHASIS -- REEXAMINE PROCESS AND PRODUCT
ORG 1	MAINTAINED GOOD STANDARDS IMPROVED TRAINING IMPROVED MEASUREMENT	LEVEL 5*	VERIFY OOD IS USED VERIFY REUSE IS HIGHER**
ORG 2	IMPROVED STANDARDS MAINTAINED GOOD TRAINING MAINTAINED GOOD MEASUREMENT	LEVEL 5*	VERIFY CLEANROOM IS USED VERIFY ERROR RATES LOWER**

*HOPEFULLY LEADS TO LOWER RISK **POSSIBLY LEADS TO LEVEL 5

SEL: SUCCESS IS MEASURED BY PRODUCT CHANGE (DOMAIN-DEPENDENT)
CMM: SUCCESS IS MEASURED BY PROCESS CHANGE (DOMAIN INDEPENDENT)

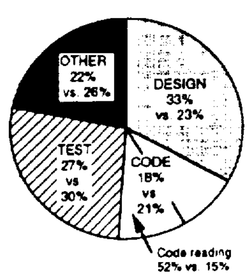
A846 013

ASSESSMENT OF CHANGES HAS PROCESS CHANGED?

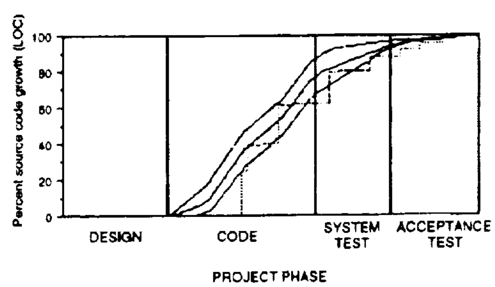
Effect of Cleanroom

IMPACT ON EFFORT DISTRIBUTION

Cleanroom vs. baseline



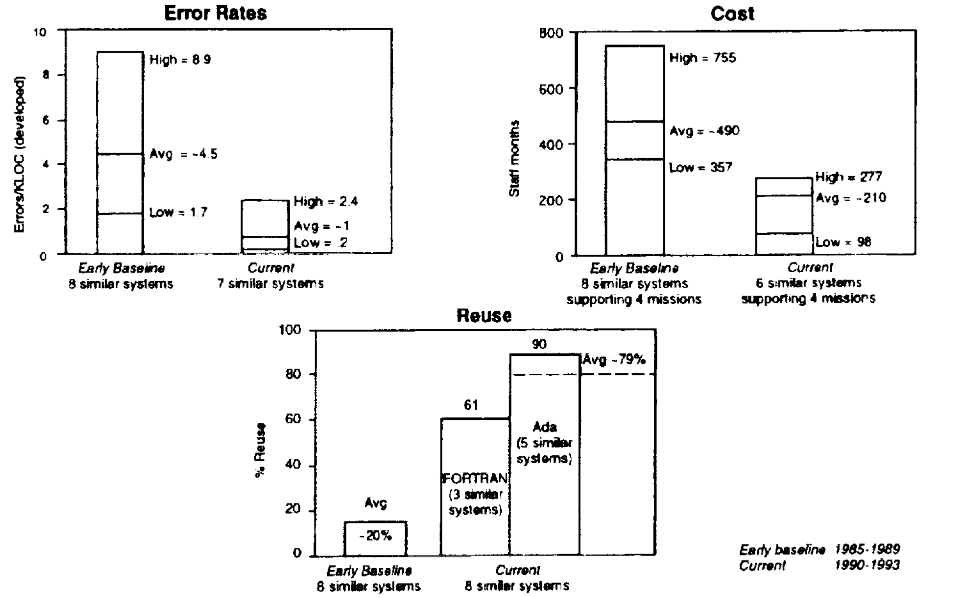
IMPACT ON SOURCE CODE (LOC) GROWTH RATE



Impact of changes are verified with process data vs. baseline

93M11U1.012

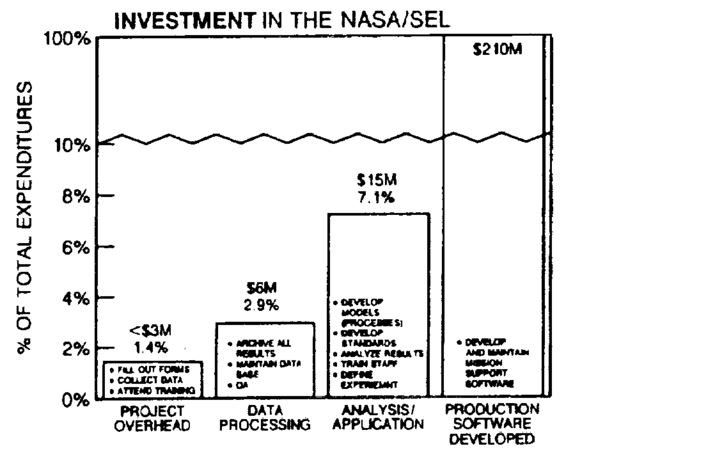
ASSESSMENT OF CHANGES
HAS PRODUCT IMPROVED?



Improvement is measured against the goals of an organization

93M11U.013

SOFTWARE PROCESS IMPROVEMENT
STEP 5 - MEASURE ROI



COST OF SEL PROCESS IMPROVEMENT ACTIVITIES TOTAL ~11% OF ALL EXPENDITURES

A846 016

MEASURING RETURN ON INVESTMENT IN THE SEL
(BASED ON CHANGES FROM MID 80s TO EARLY 90s)

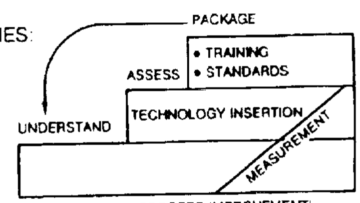
• RELIABILITY	Errors/KSLOC	down by 75%	(from 4.5 to 1)
• REUSE	Average level of code reuse	increased by 300%	(from ~20% to ~80%)
• DEVELOPMENT COST	Average mission cost*	down ~55%	(from 490 to 210 staff mos)

> Investment in product-driven goals enables direct measurement of return

*Reflects reuse change

93M11U.015

MEASURING ROI FOR PROCESS IMPROVEMENT OBSERVED ORGANIZATIONAL IMPACTS



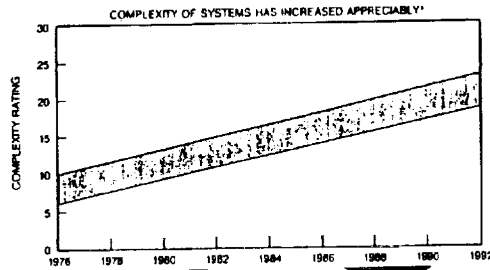
(NASA/SEL PROCESS IMPROVEMENT)

- DRIVES INTEGRATION OF PREVIOUSLY DISPARATE ACTIVITIES:
  - TRAINING
  - STANDARDS, POLICIES
  - TECHNOLOGY INSERTION
  - MEASUREMENT
- ENHANCES ROLE OF ORGANIZATIONAL ELEMENTS (CULTURAL CHANGE)
  - DEVELOPERS -- BECOME AN INTEGRAL PART OF PROCESS CHANGE
  - SOFTWARE -- PROCESS DRIVEN (LESS PEOPLE DRIVEN)
- FOCUSES ROLE OF SOFTWARE ENGINEERING RESEARCH
  - BECOMES GOAL/PROBLEM DRIVEN
  - EXISTS MEANS TO EXPERIMENT, ASSESS, ADOPT

> PROCESS IMPROVEMENT WILL DRIVE ORGANIZATIONAL EVOLUTION AND OPTIMIZE ALLOCATION OF SOFTWARE-RELATED RESOURCES

A846.17A

MEASURING ROI FOR PROCESS IMPROVEMENT



	LATE '70s-EARLY '80s	LATE '80s-EARLY '90s
CONTROL	SPIN STABILIZED	3-AXIS STABILIZED
SENSORS	1	8 TO 11
TORQUERS	1	2 TO 3
OBC	ANALOG SIMPLE CONTROL	DIGITAL AUTONOMOUS CONTROL
TELEMETRY	5	12 TO 15
DATA RATES	2.2 kb/s	32 kb/s
ACCURACY	1 degree	0.02 degree

THE NASA/SEL IS PRODUCING MORE FUNCTIONALITY, FOR MORE COMPLEX SYSTEMS, WITH HIGHER RELIABILITY, AT SIGNIFICANTLY LOWER COST.

*D. BOLAND, "A STUDY ON SIZE AND REUSE TRENDS IN ATTITUDE GROUND SUPPORT SYSTEMS (AGSSs) DEVELOPED FOR THE FLIGHT DYNAMICS DIVISION (FDD) (1976-1988)", CSC/TM-89/6031, CSC FEBRUARY 1989

A846 018